GLOBAL COMPLEXITY BOUND OF A PROXIMAL ADMM FOR LINEARLY-CONSTRAINED NONSEPARABLE NONCONVEX COMPOSITE PROGRAMMING*

WEIWEI KONG† AND RENATO D.C. MONTEIRO‡

Abstract. This paper proposes and analyzes a dampened proximal alternating direction method of multipliers (DP.ADMM) for solving linearly-constrained nonconvex optimization problems where the smooth part of the objective function is nonseparable. Each iteration of DP.ADMM consists of: (ii) a sequence of partial proximal augmented Lagrangian (AL) updates, (ii) an under-relaxed Lagrange multiplier update, and (iii) a novel test to check whether the penalty parameter of the AL function should be updated. Under a basic Slater point condition and some requirements on the dampening factor and under-relaxation parameter, it is shown that DP.ADMM obtains an approximate first-order stationary point of the constrained problem in $\mathcal{O}(\varepsilon^{-3})$ iterations for a given numerical tolerance $\varepsilon > 0$. One of the main novelties of the paper is that convergence of the method is obtained without requiring any rank assumptions on the constraint matrices.

Key words. proximal ADMM, nonseparable, nonconvex composite optimization, iteration complexity, under-relaxed update, augmented Lagrangian function

AMS subject classifications. 65K10, 90C25, 90C26, 90C30, 90C60

1. Introduction.

Consider the following composite optimization problem:

$$\min_{x\in\mathbb{R}^n} \left\{\phi(x) := f(x) + h(x) : Ax = d\right\}, \tag{1.1}$$

where h is a closed convex function, f is a (possibly) nonconvex differentiable function on the domain of h, the gradient of f is Lipschitz continuous, A is a linear operator, $d \in \mathbb{R}^\ell$ is a vector in the image of A (denoted as $\mathrm{Im}(A)$), and the following B-block structure is assumed:

$$\begin{gathered} n = n_1 + \ldots + n_B, \quad x = (x_1, \ldots, x_B) \in \mathbb{R}^{n_1} \times \cdots \times \mathbb{R}^{n_B} \\ h(x) = \sum_{t=1}^{B} h_t(x_t), \quad Ax = \sum_{t=1}^{B} A_t x_t, \end{gathered} \tag{1.2}$$

where $\{A_t\}_{t=1}^B$ is another set of linear operators and $\{h_t\}_{t=1}^B$ is another set of proper closed convex functions with compact domains.

Due to the block structure in (1.2), a popular algorithm for obtaining stationary points of (1.1) is the proximal alternating direction method of multipliers (ADMM) wherein a sequence of smaller augmented Lagrangian type subproblems is solved over $x_1, ..., x_B$ sequentially or in parallel. However, the main drawbacks of existing ADMM-type methods include: (i) strong assumptions about the structure of h; (ii) iteration

***Funding**: The first author has been supported by (i) the US Department of Energy (DOE) and UT-Battelle, LLC, under contract DE-AC05-00OR22725, (ii) the Exascale Computing Project (17-SC-20-SC), a collaborative effort of the U.S. Department of Energy Office of Science and the National Nuclear Security Administration, and (iii) the IDEaS-TRIAD Fellowship (NSF Grant CCF-1740776). The second author was partially supported by ONR Grant N00014-18-1-2077 and AFOSR Grant FA9550-22-1-0088.

Versions: v0.1 (Oct. 24, 2021), v0.2 (Dec. 9, 2021), v1.0 (Jun. 14, 2022), v2.0 (Jan. 3, 2023)

†Computer Science and Mathematics Division, Oak Ridge National Laboratory, Oak Ridge, TN, 37830. wwkong92@gmail.com

‡School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA, 30332-0205. monteiro@isye.gatech.edu

complexity bounds that scale poorly with the numerical tolerance; (iii) small stepsize parameters; or (iv) a strong rank assumption about the last block A_B that implies $\mathrm{Im}(A_B) \supseteq \{d\} \cup \mathrm{Im}(A_1) \cup \ldots \mathrm{Im}(A_{B-1})$ which we refer to as the *last block condition.*

Of the above drawbacks, (iv) is especially limiting. To illustrate this, we give a few applications where the last block condition, and hence (iv), does not hold:

- ▷ *Rank-deficient Quadratic Programming (RDQP).* It is shown in [4] that the (non-proximal) ADMM diverges on the following three-block convex RDQP:

$$\begin{aligned} &\min_{x_1,x_2,x_3,x_4} \ \frac{1}{2}x_1^2 \\ &\text{s.t.} \ \begin{pmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} 1 \\ 1 \\ 2 \end{pmatrix} x_3 + \begin{pmatrix} 1 \\ 2 \\ 2 \end{pmatrix} x_4 = 0. \end{aligned}$$

- ▷ *Distributed Finite-Sum Optimization (DFSO).* Given a positive integer B, consider:

$$\min_{x_i \in \mathbb{R}^n} \left\{ \sum_{t=1}^{B} (f_t + h_t)(x_t) : x_t - x_B = 0, \quad t = 1, \ldots, B-1 \right\} \tag{1.3}$$

  where f_i is continuously differentiable, h_t is closed convex, and ∇f_t is Lipschitz continuous for $t = 1, ..., B$. It is easy to see[1] that (1.3) is a special case of (1.1) where we have $A_s = e_s \otimes I \in \mathbb{R}^{n(B-1)\times n}$ for $s = 1, \ldots, B-1$, we have $A_B = -\mathbf{1} \otimes I \in \mathbb{R}^{n(B-1)\times n}$, and $d = 0$. Moreover, it is straightforward to show that for $s = 1, \ldots, B-1$ we have $\mathrm{Im}(A_s) \cap \mathrm{Im}(A_B) = 0$ but $\mathrm{Im}(A_s) \backslash \{0\} \neq \emptyset$, which implies that $\mathrm{Im}(A_s) \not\subseteq \mathrm{Im}(A_B)$.
- ▷ *Decentralized AC Optimal Power Control (DAC-OPF).* The convex version was first considered in [27] for the rectangular coordinate formulation, and the problem itself is considered one of the most important ones in power systems decision-making. The nonconvex version of DAC-OPF is a variant where h_t is the indicator of a convex region given by a finite number of complicated quadratic constraints and f_t is a nonconvex quadratic cost function. A discussion of the limitations induced by assuming any rank condition which implies the last block condition is given in [29].

Our goal in this paper is to develop and analyze the complexity of a proximal ADMM that removes all the drawbacks above. For a given $\theta \in (0, 1)$, its k^{th} iteration is based on the *dampened* augmented Lagrangian (AL) function given by

$$\mathcal{L}^{\theta}_{c_k}(x; p) := \phi(x) + (1-\theta)\langle p, Ax - d\rangle + \frac{c_k}{2}\|Ax - d\|^2, \tag{1.4}$$

where $c_k > 0$ is the *penalty parameter*. Specifically, it consists of the following updates: given $x^{k-1} = (x_1^{k-1}, \ldots, x_B^{k-1})$, p^{k-1} c_k, χ, and λ, sequentially $(t = 1, \ldots, B)$ compute the t^{th} block of x^k as

$$x_t^k = \operatorname*{argmin}_{u_t \in \mathbb{R}^{n_t}} \left\{ \lambda \mathcal{L}^{\theta}_{c_k}(\ldots, x_{t-1}^k, u_t, x_{t+1}^{k-1}, \ldots; p^{k-1}) + \frac{1}{2}\|u_t - x_t^{k-1}\|^2 \right\}, \tag{1.5}$$

and then update

$$p^k = (1-\theta)p^{k-1} + \chi c_k \left(Ax^k - d\right), \tag{1.6}$$

[1] Here, $e_1, \ldots, e_n$ is the standard basis for $\mathbb{R}^{B-1}$, I_n is the n-by-n identity matrix, $\mathbf{1} \in \mathbb{R}^{B-1}$ is a vector of ones, and $\otimes$ is the Kronecker product of two matrices.

where $\chi \in (0,1)$ is a suitably chosen under-relaxation parameter.

Contributions. For proper choices of the stepsize λ and a non-decreasing sequence of penalty parameters $\{c_k\}_{k\geq 1}$, it is shown that if the Slater-like condition[2]

$$\exists z_\dagger \in \operatorname{int}(\operatorname{dom} h) \ \text{such that} \ Az_\dagger = d, \tag{1.7}$$

holds, then DP.ADMM has the following features:

- ▷ for any tolerance pair $(\rho,\eta) \in \mathbb{R}^2_{++}$, it obtains a pair $(\bar{z},\bar{q})$ satisfying

  $$\operatorname{dist}(0, \nabla f(\bar{z}) + A^*\bar{q} + \partial h(\bar{z})) \leq \rho, \quad \|A\bar{z} - d\| \leq \eta \tag{1.8}$$

  in $\mathcal{O}(\max\{\rho^{-3},\eta^{-3}\})$ iterations;
- ▷ it introduces a novel approach for updating the penalty parameter c_k, instead of assuming that $c_k = c_1$ for every $k \geq 1$ and that c_1 is sufficiently large (such as in [3, 14, 15, 28, 31, 32]);
- ▷ it does not have any of the drawbacks mentioned in the sentences preceding equation (1.3).

Related Works. Since ADMM-type methods where f is convex have been well-studied in the literature (see, for example, [1, 2, 5, 6, 7, 8, 9, 10, 11, 12, 23, 24, 25]), we make no further mention of them here. Instead, we discuss below ADMM-type methods where f is nonconvex.

Letting δ_S denote the indicator function of a convex set S (see Subsection 1.1), we first present a list of common assumptions in Table 1.1.

$\mathcal{Q}$	$f(z) = \sum_{t=1}^{B} f_t(z_t)$ for subfunctions $f_t : \operatorname{dom} h_t \mapsto \mathbb{R}$.
$\mathcal{R}_0$	$\operatorname{Im}(A_B) \supseteq \{d\} \cup \operatorname{Im}(A_1) \cup \ldots \cup \operatorname{Im}(A_{B-1})$.
$\mathcal{S}$	The Slater-like assumption (1.7) holds.
$\mathcal{P}$	$h_i \equiv \delta_P$ for $i \in \{1,\ldots,B\}$, where P is a polyhedral set.
$\mathcal{F}$	A point $x^0 \in \operatorname{dom} h$ satisfying $Ax^0 = d$ is available as an input.

Table 1.1
Common nonconvex ADMM assumptions and regularity conditions.

Earlier developments on ADMM for solving nonconvex instances of (1.1) all assume that $\mathcal{R}_0$ hold, and the ones dealing with complexity establish an $\mathcal{O}(\varepsilon^{-2})$ iteration complexity, where $\varepsilon := \min\{\rho,\eta\}$. More specifically, [3, 13, 30, 31] present proximal ADMMs under the assumption $B = 2$, $h_B \equiv 0$, and assumption $\mathcal{Q}$ holds for [3, 13, 30]. Papers [14,15,20,21] present (possibly linearized) ADMMs under the assumption that $B \geq 2$, $h_B \equiv 0$, and assumption $\mathcal{Q}$ holds for [14, 20, 21].

We next discuss papers that do not assume the restrictive condition $\mathcal{R}_0$ in Table 1.1, and are based on ADMM approaches directly applicable to (1.1) or some reformulation of it. An early paper in this direction is [15], which establishes an $\mathcal{O}(\varepsilon^{-6})$ iteration-complexity bound for an ADMM-type method applied to a penalty reformulation of (1.1) that artificially satisfies $\mathcal{R}_0$. On the other hand, development of ADMM-type methods directly applicable to (1.1) is considerably more challenging and only a few works have recently surfaced (see Table 1.2 below).

We now discuss some advantages of DP.ADMM compared to the other two papers in Table 1.2. First, the method in [28] considers a small stepsize (proportional

[2]Here, $\operatorname{int} S$ denotes the interior of a set S, $\operatorname{dom}\psi$ denotes the domain of a function ψ, and A^* is the adjoint of linear operator A.

Algorithm	θ	χ	Complexity	Assumptions	Adaptive c
LPADMM [32]	0	$(0,\infty)$	None	$\mathcal{P},\mathcal{S}$	No
SDD-ADMM [28]	$(0,1]$	$[-\frac{\theta}{4},0)$	$\mathcal{O}(\varepsilon^{-4})$	$\mathcal{F}$	No
DP.ADMM	$(0,1]$	$(0,\pi_\theta]$	$\mathcal{O}(\varepsilon^{-3})$	$\mathcal{S}$	Yes

TABLE 1.2
Comparison of existing ADMM-type methods with DP.ADMM for finding ε-stationary points with $\varepsilon := \min\{\rho,\eta\}$ and $\pi_\theta = \theta^2/[2B(2-\theta)(1-\theta)]$ if $\theta \in (0,1)$ and $\pi_\theta = 1$ if $\theta = 1$.

to η^2) *linearized* proximal gradient update while DP.ADMM considers a large stepsize (proportional to the inverse of the weak-convexity constant of f) proximal point update as in (1.5). Second, the method in [28] requires a feasible initial point, i.e., a point $z_0 \in \operatorname{dom} h$ satisfying $Az_0 = d$, while DP.ADMM only requires that the initial point be in $\operatorname{dom} h$. Third, the methods in [28, 32] both require certain hyperparameters (the penalty parameter in [28] and an interpolation parameter in [32]) to be chosen in a range that is hard to compute, while DP.ADMM only requires its main hyperparameter pair (χ,θ) to satisfy a simple inequality (see (2.6)). Moreover, [28] does not specify an easily implementable rule for updating its method's penalty parameter, while DP.ADMM does. Fourth, convergence of the method in [32] requires h being the indicator of a polyhedral set, whereas DP.ADMM applies to any closed convex function h. Fifth, in contrast to [28] and this work, [32] does not give a complexity bound for its proposed method. Finally, [28] considers an unusual negative stepsize for its Lagrange multiplier update — which justifies its moniker "scaled dual descent ADMM" — whereas DP.ADMM considers a positive stepsize.

Organization. Subsection 1.1 presents some basic definitions and notation. Section 2 presents the proposed DP.ADMM in two subsections. The first one precisely describes the problem of interest, while the second one states the static and dynamic DP.ADMM variants and their iteration complexities. Section 3 and 4 present the main properties of the static and dynamic DP.ADMM, respectively. Section 5 presents some preliminary numerical experiments. Section 6 gives some concluding remarks. Finally, the end of the paper contains several appendices.

1.1. Notation and Basic Definitions. Let $\mathbb{R}_+$ denote the set of nonnegative real numbers, and let $\mathbb{R}_{++}$ denote the set of positive real numbers. Let $\mathbb{R}_n$ denote the n-dimensional Hilbert space with inner product and associated norm denoted by $\langle\cdot,\cdot\rangle$ and $\|\cdot\|$, respectively. The direct sum (or Cartesian product) of a set of sets $\{S_i\}_{i=1}^n$ is denoted by $\prod_{i=1}^n S_i$.

The smallest positive singular value of a nonzero linear operator $Q:\mathbb{R}^n \to \mathbb{R}^l$ is denoted by σ_Q^+. For a given closed convex set $X \subset \mathbb{R}^n$, its boundary is denoted by ∂X and the distance of a point $x \in \mathbb{R}^n$ to X is denoted by $\operatorname{dist}_X(x)$. The indicator function of X at a point $x \in \mathbb{R}^n$ is denoted by $\delta_X(x)$ which has value 0 if $x \in X$ and $+\infty$ otherwise. For every $z > 0$ and positive integer b, we denote $\log_b^+(z) := \max\{1, \lceil \log_b(z) \rceil\}$.

The domain of a function $h:\mathbb{R}^n \to (-\infty,\infty]$ is the set $\operatorname{dom} h := \{x \in \mathbb{R}^n : h(x) < +\infty\}$. Moreover, h is said to be proper if $\operatorname{dom} h \neq \emptyset$. The set of all lower semi-continuous proper convex functions defined in $\mathbb{R}^n$ is denoted by $\overline{\operatorname{Conv}}\, \mathbb{R}^n$. The set of functions in $\overline{\operatorname{Conv}}\, \mathbb{R}^n$ which have domain $Z \subseteq \mathbb{R}^n$ is denoted by $\overline{\operatorname{Conv}}\, Z$. The ε-subdifferential of a proper function $h:\mathbb{R}^n \to (-\infty,\infty]$ is defined by

$$\partial_\varepsilon h(z) := \{u \in \mathbb{R}^n : h(z') \geq h(z) + \langle u, z'-z\rangle - \varepsilon, \quad \forall z' \in \mathbb{R}^n\} \tag{1.9}$$

for every $z \in \mathbb{R}^n$. The classic subdifferential, denoted by $\partial h(\cdot)$, corresponds to $\partial_0 h(\cdot)$. The normal cone of a closed convex set C at $z \in C$, denoted by $N_C(z)$, is defined as

$$N_C(z) := \{\xi \in \mathbb{R}^n : \langle \xi, u - z\rangle \le \varepsilon, \quad \forall u \in C\}.$$

If ψ is a real-valued function which is differentiable at $\bar{z} \in \mathbb{R}^n$, then its affine approximation $\ell_\psi(\cdot, \bar{z})$ at $\bar{z}$ is given by

$$\ell_\psi(z; \bar{z}) := \psi(\bar{z}) + \langle \nabla \psi(\bar{z}), z - \bar{z}\rangle \quad \forall z \in \mathbb{R}^n. \tag{1.10}$$

If $z = (x, y)$ then $f(x, y)$ is equivalent to $f(z) = f((x, y))$.

Iterates of a scalar quantity have their iteration number appear as a subscript, e.g., c_ℓ, while non-scalar quantities have this number appear as a superscript, e.g., v^k, and $\hat{p}^\ell$. For variables with multiple blocks, the block number appears as a subscript, e.g., x_t^k and v_t^k. Finally, we define the following norm for any quantity $u = (u_1, \ldots, u_B)$ following a block structure as in (1.2):

$$\|u\|_\dagger = \|(u_1, \ldots, u_B)\|_\dagger := \sum_{t=1}^{B} \|u_t\|. \tag{1.11}$$

2. Alternating Direction Method of Multipliers. This section contains two subsections. The first one precisely describes the problem of interest and its underlying assumptions, while the second one presents the DP.ADMM and its corresponding iteration complexity.

2.1. Problem of Interest. This subsection presents the problem of interest and the assumptions underlying it.

Denote the aggregated quantities

$$\begin{aligned} x_{<t} &:= (x_1, \ldots, x_{t-1}), \quad x_{>t} := (x_{t+1}, \ldots, x_B), \\ x_{\le t} &:= (x_{<t}, x_t), \quad x_{\ge t} := (x_t, x_{>t}), \end{aligned} \tag{2.1}$$

for every $x = (x_1, \ldots, x_B) \in \mathcal{H}$. Our problem of interest is finding approximate stationary points of (1.1) under the following assumptions:

(A1) for every $t = 1, \ldots, B$, we have $h_t \in \overline{\mathrm{Conv}}\, \mathbb{R}^{n_t}$ and $\mathcal{H}_t := \mathrm{dom}\, h_t$ is compact;

(A2) $A \not\equiv 0$ and $\mathcal{F} := \{x \in \mathcal{H} : Ax = d\} \ne \emptyset$ where $\mathcal{H} := \mathcal{H}_1 \times \cdots \times \mathcal{H}_B$;

(A3) h in (1.2) is K_h-Lipschitz continuous on $\mathcal{H}$ for some $K_h \ge 0$;

(A4) for every $t = 1, \ldots, B$, there exists $m_t \ge 0$ such that

$$f(x_{<t}, \cdot, x_{>t}) + \delta_{\mathcal{H}_t}(\cdot) + \frac{m_t}{2}\|\cdot\|^2 \text{ is convex for all } x \in \mathcal{H}; \tag{2.2}$$

(A5) f is differentiable on $\mathcal{H}$ and, for every $t = 1, \ldots, B-1$, there exists $M_t \ge 0$ such that

$$\|\nabla_{x_t} f(x_{\le t}, \tilde{x}_{>t}) - \nabla_{x_t} f(x_{\le t}, x_{>t})\| \le M_t \|\tilde{x}_{>t} - x_{>t}\| \quad \forall x, \tilde{x} \in \mathcal{H}; \tag{2.3}$$

(A6) there exists $z_\dagger \in \mathcal{F}$ such that $d_\dagger := \mathrm{dist}_{\partial\mathcal{H}}(z_\dagger) > 0$.

We now give a few remarks about the above assumptions. First, in view of the fact that $\mathcal{H}$ is compact, the following scalars are bounded:

$$\begin{aligned} D_\dagger &:= \sup_{z \in \mathcal{H}} \|z - z_\dagger\|, \quad G_f := \sup_{x \in \mathcal{H}} \|\nabla f(x)\|, \\ \underline{\phi} &:= \inf_{x \in \mathcal{H}} \phi(x), \quad \overline{\phi} := \sup_{x \in \mathcal{H}} \phi(x). \end{aligned} \tag{2.4}$$

Second, if f is a separable function, i.e., it is of the form $f(z) = f_1(z_1) + \cdots + f_B(z_B)$, then each M_t can be chosen to be zero. Third, any function h given by (1.2) such that each h_t for $t = 1, \ldots, B$ has the form $h_t = \tilde{h}_t + \delta_{Z_t}$, where $\tilde{h}_t$ is a finite everywhere Lipschitz continuous convex function and Z_t is a compact convex set, clearly satisfies condition (A3) for some K_h.

For a given tolerance pair (ρ, η), we define a (ρ, η)-stationary pair of (1.1) as being a pair $(\bar{z}, \bar{q}) \in \mathcal{H} \times \mathbb{R}^\ell$ satisfying (1.8). It is well known that the first-order necessary condition for a point $z \in \mathcal{H}$ to be a local minimum of (1.1) is that there exists $q \in \mathbb{R}^\ell$ such that the stationary conditions

$$0 \in \nabla f(z) + A^* q + \partial h(z), \quad Az = d$$

hold. Hence, the requirements in (1.8) can be viewed as a direct relaxation of the above stationary conditions. For ease of future reference, we consider the following problem.

Problem $\mathcal{S}_{\rho,\eta}$: Find a (ρ, η)-stationary pair $(\bar{z}, \bar{q})$ satisfying (1.8).

We now make three remarks about Problem $\mathcal{S}_{\rho,\eta}$. First, $(\bar{z}, \bar{q})$ is a solution of Problem $\mathcal{S}_{\rho,\eta}$ if and only if there exists a residual $\bar{v} \in \mathbb{R}^n$ such that

$$\bar{v} \in \nabla f(\bar{z}) + A^* \bar{q} + \partial h(\bar{z}), \quad \|\bar{v}\| \le \rho, \quad \|A\bar{z} - d\| \le \eta. \tag{2.5}$$

Second, condition (2.5) has been considered in many previous works (e.g., see [16, 17, 18, 19, 22]). Third, in the case where $\|\cdot\| = \|\cdot\|_2$ and $\rho = \eta$, the stationarity condition in (1.8) implies the stationarity condition of the papers [15,28] in Table 1.2. Specifically, [15, Definition 3.6] and [28, Definition 3.3] consider a pair $(z, q) \in \mathcal{H} \times \mathbb{R}^\ell$ to be an ε-stationary pair if it satisfies

$$\operatorname{dist}(0, \nabla_{z_t} f(z_1, \ldots, z_B) + A_t^* q + \partial h_t(z_t)) \le \varepsilon, \quad \|Az - d\| \le \varepsilon,$$

for every $t = 1, \ldots, B$.

In the following subsection, we present a method (Algorithm 2.1) that computes a triple $(\bar{z}, \bar{q}, \bar{v})$ satisfying (2.5), and hence which guarantees that $(\bar{z}, \bar{q})$ is a solution of Problem $\mathcal{S}_{\rho,\eta}$.

2.2. DP.ADMM. We present DP.ADMM in two parts. The first part presents a static version of DP.ADMM which either (i) stops with a solution of Problem $\mathcal{S}_{\rho,\eta}$ or (ii) signals that its penalty parameter is too small. The second part presents the (dynamic) DP.ADMM that repeatedly invokes the static version on an increasing sequence of penalty parameters.

Both versions of DP.ADMM make use of the following condition on (χ, θ):

$$2\chi B(2 - \theta)(1 - \theta) \le \theta^2, \quad (\chi, \theta) \in (0, 1]^2. \tag{2.6}$$

For ease of reference and discussion, the pseudocode for the static DP.ADMM is given in Algorithm 2.1 below. Notice that the classic proximal ADMM iteration

$$x_t^k = \underset{u^t \in \mathbb{R}^{n_t}}{\operatorname{argmin}} \left\{ \lambda \mathcal{L}_c^0(x_{<t}^k, u_t, x_{>t}^{k-1}; p^{k-1}) + \frac{1}{2}\|u_t - x_t^{k-1}\|^2 \right\}, \quad t = 1, \ldots, B,$$

$$p^k = p^{k-1} + c\left(Ax^k - d\right),$$

Algorithm 2.1 Static DP.ADMM

`Input`: $x^0 \in \mathcal{H}$, $p^0 \in A(\mathbb{R}^n)$, $\lambda \in (0, 1/(2m)]$, $c > 0$;
`Require`: m as in (2.7), $(\rho, \eta) \in \mathbb{R}^2_{++}$, (χ, θ) as in (2.6)

for $k \leftarrow 1, 2, \ldots$ **do**
  `STEP 1` (prox update):
  for $t \leftarrow 1, 2, \ldots, B$ **do**
    $x_t^k \leftarrow \operatorname{argmin}_{u_t \in \mathbb{R}^{n_t}} \left\{ \lambda \mathcal{L}_c^\theta(x_{<t}^k, u_t, x_{>t}^{k-1}; p^{k-1}) + \frac{1}{2}\|u_t - x_t^{k-1}\|^2 \right\}$
  $q^k \leftarrow (1-\theta)p^{k-1} + c(Ax^k - d)$
  `STEP 2a` (successful termination check):
  for $t \leftarrow 1, 2, \ldots, B$ **do**
    $\delta_t^k \leftarrow \nabla_{x_t} f(x_{\le t}^k, x_{>t}^k) - \nabla_{x_t} f(x_{\le t}^k, x_{>t}^{k-1})$
    $v_t^k \leftarrow \delta_t^k + cA_t^* \sum_{s=t+1}^B A_s(x_s^k - x_s^{k-1}) - \frac{1}{\lambda}(x_t^k - x_t^{k-1})$
  if $\|v^k\| \le \rho$ **and** $\|Ax^k - d\| \le \eta$ **then**
    return (x^k, p^k, q^k, v^k)
  `STEP 2b` (unsuccessful termination check):
  if $k \equiv 0 \bmod 2$ and $k \ge 3$ **then**
    $\mathcal{S}_k^{(v)} \leftarrow \frac{2}{k+2}\sum_{i=k/2}^k \|v^i\|$
    $\mathcal{S}_k^{(f)} \leftarrow \frac{2}{k+2}\sum_{i=k/2}^k \|Ax^i - d\|$
    if $\frac{1}{\rho} \cdot \mathcal{S}_k^{(v)} + \frac{1}{\eta}\sqrt{\frac{c^3}{k}} \cdot \mathcal{S}_k^{(f)} \le 1$ **then**
      return (x^k, p^k, q^k, v^k)
  `STEP 3` (multiplier update):
  $p^k \leftarrow (1-\theta)p^{k-1} + \chi c(Ax^k - d)$

corresponds to the case of $(\chi, \theta) = (1, 0)$, where $c \ge 1$ is a fixed penalty parameter.

The next result describes the iteration complexity and some useful technical properties of Algorithm 2.1. Its proof is given in Section 3.3, and it uses three sets of scalars. The first set is independent of (c, p^0) and is given by

$$
\begin{aligned}
&M := \max_{1\le t\le B} M_t, \quad m := \max_{1\le t\le B} m_t, \quad \Delta_\phi := \overline{\phi} - \underline{\phi}, \quad \kappa_0 := \frac{2B^2(\lambda M + 1)}{\sqrt{\lambda}}, \\
&\kappa_1 := \frac{\chi\|A\|D_\dagger}{\theta}, \quad \kappa_2 := \frac{1}{\theta}\left[1 + \frac{2\chi D_\dagger(K_h + G_f)}{\theta d_\dagger \sigma_A^+}\right] + 1, \\
&\kappa_3 := \frac{108\kappa_2^2}{\chi^2}, \quad \kappa_4 := \frac{\theta d_\dagger \sigma_A^+}{\chi D_\dagger}, \quad \kappa_5 := 8(B-1)\|A\|_\dagger^2, \quad \kappa_6 := 3 + \frac{8\kappa_0^2\Delta_\phi}{\kappa_4^2}.
\end{aligned} \tag{2.7}
$$

where $(G_f, D_\dagger, \overline{\phi}, \underline{\phi})$, K_h, and (m_t, M_t) are as in (2.4), (A3), and (A4). The second set is dependent on a given lower bound $\underline{c}$ on c and is given by

$$
\tilde{\kappa}_{\underline{c}}^{(0)} := 2\left(\sqrt{\Delta_\phi} + \frac{5\kappa_2}{\chi\sqrt{\underline{c}}}\right), \quad \tilde{\kappa}_{\underline{c}}^{(1)} := 3\kappa_5[\tilde{\kappa}_{\underline{c}}^{(0)}]^2, \quad \tilde{\kappa}_{\underline{c}}^{(2)} := 3\kappa_0^2[\tilde{\kappa}_{\underline{c}}^{(0)}]^2. \tag{2.8}
$$

The third set is dependent on a given upper bound $\mathcal{R}$ on $\|p^0\|/c$ and is given by

$$\begin{aligned} \xi_{\mathcal{R}}^{(0)} &:= \frac{8}{\kappa_4^2}\left[\frac{9\kappa_0^2(\mathcal{R}+\kappa_1)^2}{\chi^2}+\kappa_5\Delta_\phi\right]+(1-\theta)(\mathcal{R}+\kappa_1), \\ \xi_{\mathcal{R}}^{(1)} &:= \frac{72\kappa_5(\mathcal{R}+\kappa_1)^2}{\chi^2\kappa_4^2}. \end{aligned} \tag{2.9}$$

PROPOSITION 2.1. *Let $\mathcal{R}\geq 0$ and $\underline{c}>0$ be given, and assume that the pair (c,p^0) given to Algorithm 2.1 satisfies*

$$\|p_0\|\leq c\mathcal{R}, \quad c\geq \underline{c}. \tag{2.10}$$

Then, the following statements hold about the call to Algorithm 2.1:

(a) it terminates in a number of iterations bounded by

$$\mathcal{T}_c(\rho,\eta\,|\,\underline{c},\mathcal{R}) := 48\left(\left\{\kappa_6+\frac{\tilde{\kappa}_{\underline{c}}^{(1)}}{\rho^2}\right\}+\left\{\xi_{\mathcal{R}}^{(0)}+\frac{\kappa_3}{\eta^2}+\frac{\tilde{\kappa}_{\underline{c}}^{(2)}}{\rho^2}\right\}c+\xi_{\mathcal{R}}^{(1)}c^2\right), \tag{2.11}$$

where (κ_3,κ_6), $(\tilde{\kappa}_{\underline{c}}^{(1)},\tilde{\kappa}_{\underline{c}}^{(2)})$, and $(\xi_{\mathcal{R}}^{(0)},\xi_{\mathcal{R}}^{(1)})$ are as in (2.7), (2.8), and (2.9), respectively;

(b) if it terminates successfully in Step 2a, then the first and third components of its output quadruple $(\bar{z},\bar{p},\bar{q},\bar{v})$ solve Problem $\mathcal{S}_{\rho,\eta}$;

(c) if c satisfies

$$c\geq \hat{c}(\rho,\eta\,|\,\underline{c},\mathcal{R}) := \frac{1}{\underline{c}^2}\left[\mathcal{T}_{\underline{c}}(1,1\,|\,\underline{c},\mathcal{R})+\frac{\sqrt{\underline{c}^3\cdot\mathcal{T}_{\underline{c}}(1,1\,|\,\underline{c},\mathcal{R})}}{\min\{\rho,\eta\}}\right], \tag{2.12}$$

where $\mathcal{T}_c(\rho,\eta\,|\,\underline{c},\mathcal{R})$ is as in (a), then it must terminate successfully.

We now make some remarks about Proposition 2.1. First, statement (c) implies that Algorithm 2.1 terminates successfully if its penalty parameter c is sufficiently large, i.e., $c=\Omega(\varepsilon^{-1})$ where $\varepsilon:=\min\{\rho,\eta\}$. Moreover, if a penalty parameter c satisfying (2.12) and the condition that $c=\mathcal{O}(\varepsilon^{-1})$ is known, then it follows from Proposition 2.1(a) that the iteration complexity of Algorithm 2.1 for finding a solution of Problem $\mathcal{S}_{\rho,\eta}$ is $\mathcal{O}(\varepsilon^{-3})$.

Since a penalty parameter c as in the above paragraph is nearly impossible to compute, we next present an adaptive method, namely, Algorithm 2.2 below, which adaptively increases the penalty parameter c, and whose overall number of iterations is also $\mathcal{O}(\varepsilon^{-3})$.

Some comments about Algorithm 2.2 are in order. First, it employs a "warm-start" type strategy for calling Algorithm 2.1 at each iteration ℓ. Specifically, the input of the ℓ^{th} to Algorithm 2.1 is the pair $(\bar{z}^{\ell-1},\bar{p}^{\ell-1})$ output by the previous call to Algorithm 2.1. Second, the initial penalty parameter c_1 can be chosen to be any positive scalar, in contrast to many of the methods listed in Section 1 where this parameter must be chosen sufficiently large. Third, the initial point $\bar{z}^0$ only needs to be in the domain of h and need not be feasible or near feasible. Finally, while the initial Lagrange multiplier $\bar{p}^0$ is chosen to be zero, the analysis in this paper can be carried out for any $\bar{p}^0\in A(\mathbb{R}^n)$, at the cost of more complicated complexity bounds.

The next result, whose proof is given in Section 4, gives the complexity of Algorithm 2.2 in terms of the total number of iterations of Algorithm 2.1 across all of its calls.

Algorithm 2.2 DP.ADMM

`Input:` $\bar{z}^0 \in \mathcal{H}$, $\lambda \in (0, 1/(2m)]$, $c_1 > 0$

`Require:` m as in (2.7), $(\rho, \eta) \in (0,1)^2$, (χ, θ) as in (2.6)

1: $\bar{p}^0 \leftarrow 0$
2: **for** $\ell \leftarrow 1, 2, \ldots$ **do**
3: **call** Algorithm 2.1 with inputs $(x^0, p^0, \lambda, c) = (\bar{z}^{\ell-1}, \bar{p}^{\ell-1}, \lambda, c_\ell)$ and parameters m, (ρ, η), and (χ, θ) to obtain an output quadruple $(\bar{z}^\ell, \bar{p}^\ell, \bar{q}^\ell, \bar{v}^\ell)$
4: **if** $\|\bar{v}^\ell\| \le \rho$ **and** $\|A\bar{z}^\ell - d\| \le \eta$ **then**
5: **return** $(\bar{z}^\ell, \bar{q}^\ell)$
6: $c_{\ell+1} \leftarrow 2c_\ell$

THEOREM 2.2. *Define the scalars*

$$T_1 := \mathcal{T}_{c_1}(1, 1 \,|\, c_1, 2\kappa_1), \quad \varepsilon := \min\{\rho, \eta\}, \tag{2.13}$$

where κ_1 and $\mathcal{T}_c(\cdot,\cdot\,|\,\cdot,\cdot)$ are as in (2.7) and (2.11), respectively. Then, Algorithm 2.2 stops and outputs a pair that solves Problem $\mathcal{S}_{\rho,\eta}$ in a number of iterations of Algorithm 2.1 bounded by

$$T_1 \left(2E_0^2 + \frac{E_0 + 2E_1^2}{\varepsilon^2} + \frac{E_1}{\varepsilon^3} \right) \tag{2.14}$$

where

$$E_0 := 2\left(1 + \frac{T_1^2}{c_1^3}\right), \quad E_1 := 2\sqrt{\frac{T_1}{c_1^3}}. \tag{2.15}$$

Since $T_1 = \mathcal{O}(c_1^{-1})$ in view of (2.11) and (2.13), it follows from (2.14) and (2.15) that if $c_1^{-1} = \mathcal{O}(1)$, then the overall complexity of Algorithm 2.2 is $\mathcal{O}(\varepsilon^{-3})$.

3. Analysis of Algorithm 2.1.

This section presents the main properties of Algorithm 2.1, and it contains three subsections. More specifically, the first (resp., second) subsection establishes some key bounds on the ergodic means of the sequences $\{\|v^k\|\}_{k\ge0}$ and $\{\|Ax^k - d\|\}_{k\ge0}$ (resp., the sequence $\{\|p_k\|\}_{k\ge0}$). The third one proves Proposition 2.1.

Throughout this section, we let $\{(v^i, x^i, p^i, q^i)\}_{i=1}^k$ denote the iterates generated by Algorithm 2.1 up to and including the k^{th} iteration for some $k \ge 3$. Moreover, for every $i \ge 1$ and $(\chi, \theta) \in \mathbb{R}^2_{++}$ satisfying (2.6), we make use of the following useful constants and shorthand notation

$$\begin{aligned} a_\theta &= \theta(1-\theta), \quad b_\theta := (2-\theta)(1-\theta), \\ \gamma_\theta &:= \frac{(1 - 2B\chi b_\theta) - (1-\theta)^2}{2\chi}, \quad f^i := Ax^i - d, \end{aligned} \tag{3.1}$$

the aggregated quantities in (2.1), and the averaged quantities

$$S_{j,k}^{(p)} := \frac{\sum_{i=j}^k \|p^i\|}{k-j+1}, \quad S_{j,k}^{(v)} := \frac{\sum_{i=j}^k \|v^i\|}{k-j+1}, \quad S_{j,k}^{(f)} := \frac{\sum_{i=j}^k \|f^i\|}{k-j+1}. \tag{3.2}$$

for every $j = 1, \ldots, k$. Notice that $\gamma_\theta \ge \theta/\chi$ in view of (2.6). We also denote Δy^i to be the difference of iterates for any variable y at iteration i, i.e.,

$$\Delta y^i \equiv y^i - y^{i-1}. \tag{3.3}$$

3.1. Properties of the Key Residuals. This subsection presents bounds on the residuals $\{\|v^i\|\}_{i=2}^k$ and $\{\|f^i\|\}_{i=2}^k$ generated by Algorithm 2.1. These bounds will be particularly helpful for proving Proposition 2.1 in Subsection 3.3.

The first result presents some key properties about the generated iterates.

LEMMA 3.1. *For $i = 1, \ldots, k$,*

(a) $f^i = \left[p^i - (1-\theta)p^{i-1}\right]/(\chi c)$;

(b) $v^i \in \nabla f(x^i) + A^* q^i + \partial h(x^i)$ *and*

$$\|v^i\| \le B\left(M + \frac{1}{\lambda}\right)\|\Delta x^i\|_\dagger + c\|A\|_\dagger \sum_{t=2}^{B} \|A_t \Delta x_t^i\|, \tag{3.4}$$

where $\|\cdot\|_\dagger$ is as in (1.11).

Proof. (a) This is immediate from step 3 of Algorithm 2.1 and the definition of f^i in (3.1).

(b) We first prove the required inclusion. The optimality of x_t^k in Step 1 of Algorithm 2.1, and assumption (A4), imply that

$$\begin{aligned}
0 &\in \partial\left[\mathcal{L}_c^\theta(x_{<t}^i, \cdot, x_{>t}^{i-1}; p^{i-1}) + \frac{1}{2\lambda}\|\cdot - x_k^{i-1}\|^2\right](x^i) \\
&= \nabla_{x_t} f(x_{\le t}^i, x_{>t}^{i-1}) + A_t^* \left[(1-\theta)p^{i-1} + c[A(x_{\le t}^i, x_{>t}^{i-1}) - d]\right] + \partial h_t(x_t^i) + \frac{1}{\lambda}\Delta x_t^i \\
&= \nabla_{x_t} f(x_{\le t}^i, x_{>t}^{i-1}) + A_t^*\left(q^i - c\sum_{s=t+1}^{B} A_s \Delta x_s^i\right) + \partial h_t(x_t^i) + \frac{1}{\lambda}\Delta x_t^i \\
&= \nabla_{x_t} f(x^i) + A_t^* q^i + \partial h_t(x_t^i) - v_t^i.
\end{aligned}$$

for every $1 \le t \le B$. Hence, the inclusion holds. To show the inequality, let $1 \le t \le B$ be fixed and use the triangle inequality, the definition of v_t^i, and assumption (A5) to obtain

$$\begin{aligned}
\|v_t^i\| &\le \|\nabla_{x_t} f(x_{\le t}^i, x_{>t}^i) - \nabla_{x_t} f(x_{\le t}^i, x_{>t}^{i-1})\| + c\sum_{s=t+1}^{B} \|A_t^* A_s \Delta x_s^i\| + \frac{1}{\lambda}\|\Delta x_t^i\| \\
&\le M_t \|x_{>t}^i - x_{>t}^{i-1}\| + c\|A_t\| \sum_{s=t+1}^{B} \|A_s \Delta x_s^i\| + \frac{1}{\lambda}\|\Delta x_t^i\| \\
&\le \left(M + \frac{1}{\lambda}\right)\sum_{s=t}^{B} \|\Delta x_s^i\| + c\|A_t\| \sum_{t=2}^{B} \|A_t \Delta x_t^i\|.
\end{aligned}$$

Summing the above bound from $t = 1$ to B, and using the definition of M in (2.7) and the triangle inequality, we conclude that

$$\begin{aligned}
\|v^i\| \le \sum_{t=1}^{B} \|v_t^i\| &\le \left(M + \frac{1}{\lambda}\right)\sum_{t=1}^{B}\sum_{s=t}^{B} \|\Delta x_s^i\| + c\|A\|_\dagger \sum_{t=2}^{B} \|A_t \Delta x_t^i\| \\
&\le B\left(M + \frac{1}{\lambda}\right)\|\Delta x^i\|_\dagger + c\|A\|_\dagger \sum_{t=2}^{B} \|A_t \Delta x_t^i\|. \qquad \square
\end{aligned}$$

Notice that part (b) of the above result implies that $(\bar{x}, \bar{v}, \bar{p}) = (x^i, v^i, q^i)$ satisfies the inclusion in (2.5). Hence, if $\|v^i\|$ and $\|f^i\|$ are sufficiently small at some iteration

i, then Algorithm 2.1 clearly returns a solution of Problem $\mathcal{S}_{\rho,\eta}$ at iteration i, i.e., Proposition 2.1(b) holds. However, to understand when Algorithm 2.1 terminates, we will need to develop more refined bounds on $\|v_i\|$ and $\|f_i\|$.

To begin, we present some relations between the perturbed augmented Lagrangian $\mathcal{L}_c^\theta(\cdot;\cdot)$ and the iterates $\{(x^i,p^i)\}_{i=1}^k$. For conciseness, its proof is given in Appendix A.

Lemma 3.2. *For $i = 1,\ldots,k$,*

(a) $\mathcal{L}_c^\theta(x^i;p^i) - \mathcal{L}_c^\theta(x^i;p^{i-1}) = b_\theta\|\Delta p^i\|^2/(2\chi c) + a_\theta\left(\|p^i\|^2 - \|p^{i-1}\|^2\right)/(2\chi c)$;

(b) $\mathcal{L}_c^\theta(x^i;p^{i-1}) - \mathcal{L}_c^\theta(x^{i-1};p^{i-1}) \le -\|\Delta x^i\|^2/(2\lambda) - c\sum_{t=1}^B \|A_t\Delta x_t^i\|^2/2$;

(c) if $i \ge 2$, it holds that

$$\frac{b_\theta}{2\chi c}\|\Delta p^i\|^2 - \frac{c}{4}\sum_{t=1}^B \|A_t\Delta x_t^i\|^2 \le \frac{\gamma_\theta}{4B\chi c}\left(\|\Delta p^{i-1}\|^2 - \|\Delta p^i\|^2\right). \tag{3.5}$$

The next result uses the above relations to establish a bound on the quantities in the right-hand-side of (3.4).

Lemma 3.3. *For $j = 1,\ldots,k$,*

$$\sum_{i=j+1}^k \|v^i\|^2 \le (\kappa_0^2 + \kappa_5 c)\left[\Psi_j(c) - \Psi_k(c)\right], \tag{3.6}$$

where (κ_0,κ_5) is as in (2.7), and denoting (a_θ,γ_θ) and as in (3.1), we have

$$\Psi_i(c) := \mathcal{L}_c^\theta(x^i;p^i) - \frac{a_\theta}{2\chi c}\|p^i\|^2 + \frac{\gamma_\theta}{4B\chi c}\|\Delta p^i\|^2 \quad \forall i \ge 1. \tag{3.7}$$

Proof. Using the inequality $\|z\|_1^2 \le n\|z\|_2^2$ for $z \in \mathbb{R}^n$ and (3.4), we first have that

$$\begin{aligned}
\sum_{i=j+1}^k \|v^i\|^2 &\overset{(3.4)}{\le} \sum_{i=j+1}^k \left[B\left(M + \frac{1}{\lambda}\right)\|\Delta x^i\|_\dagger + c\|A\|_\dagger \sum_{t=2}^B \|A_t\Delta x_t^i\|\right]^2 \\
&\le \sum_{i=j+1}^k 2B^2\left(M + \frac{1}{\lambda}\right)^2\|\Delta x^i\|_\dagger^2 + c^2\|A\|_\dagger^2\left(\sum_{t=2}^B \|A_t\Delta x_t^i\|\right)^2 \\
&\le \sum_{i=j+1}^k 2B^4\left(M + \frac{1}{\lambda}\right)^2\|\Delta x^i\|^2 + 2(B-1)c^2\|A\|_\dagger^2 \sum_{t=2}^B \|A_t\Delta x_t^i\|^2 \\
&\le (\kappa_0^2 + \kappa_5 c)\sum_{i=j+1}^k \left[\frac{1}{2\lambda}\|\Delta x^i\| + \frac{c}{4}\sum_{t=2}^B \|A_t\Delta x_t^i\|^2\right]. \qquad (3.8)
\end{aligned}$$

Combining Lemma 3.2(a)–(c), the definition of Ψ_θ^i, and the bound $(a+b)^2 \le 2a^2 + 2b^2$ for $a,b \in \mathbb{R}_+$, we also have that

$$\begin{aligned}
&\frac{1}{2\lambda}\|\Delta x^i\|^2 + \frac{c}{4}\sum_{t=2}^B \|A_t\Delta x^i\|^2 \\
&\overset{\text{L.3.2(a)-(b)}}{\le} \mathcal{L}_c^\theta(x^{j-1};p^{j-1}) - \mathcal{L}_c^\theta(x^j;p^j) + \frac{a_\theta}{2\chi c}\Delta_{p,j}^{(2)} + \frac{b_\theta}{2\chi c}\|\Delta p^i\|^2 - \frac{c}{4}\sum_{t=1}^B \|A_t\Delta x_t^i\|^2 \\
&\overset{\text{L.3.2(c)}}{\le} \mathcal{L}_c^\theta(x^{j-1};p^{j-1}) - \mathcal{L}_c^\theta(x^j;p^j) + \frac{a_\theta}{2\chi c}\Delta_{p,j}^{(2)} + \frac{\gamma_\theta}{4B\chi c}\left(\|\Delta p^{i-1}\|^2 - \|\Delta p^i\|^2\right)
\end{aligned}$$

$$= \quad \Psi_{i-1}(c) - \Psi_i(c),$$

where $\Delta^{(2)}_{p,j} := \|p^j\|^2 - \|p^{j-1}\|^2$. Consequently, summing the above inequality from $i = j+1$ to k, and combining the resulting inequality with (3.8), yields the desired bound. ∎

We now bound the quantity on the right-hand-side of (3.6)

Lemma 3.4. *For any $j \geq 1$ and $k \geq 1$,*
(a) $\mathcal{L}^\theta_c(x^j; p^j) \leq \phi(x^j) + 3(\|p^j\|^2 + \|p^{j-1}\|^2)/(\chi^2 c)$;
(b) $\mathcal{L}^\theta_c(x^k; p^k) \geq \phi(x^k) - \|p^k\|^2/(2c)$;
(c) it holds that

$$\Psi_j(c) - \Psi_k(c) \leq \Delta_\phi + 4\left(\frac{\|p^j\|^2 + \|p^{j-1}\|^2 + \|p^k\|^2}{\chi^2 c}\right), \tag{3.9}$$

where $\Psi_i(\cdot)$ and Δ_ϕ are as in (3.6) and (2.7), respectively.

Proof. (a)–(b) See Appendix A.

(c) Using parts (a)–(b), the fact that $a_\theta \in (0,1)$ and $(\chi, \theta) \in (0,1)^2$, the relation $(a+b)^2 \leq 2a^2 + 2b^2$ for $a, b \in \mathbb{R}_+$, and the bound $\gamma_\theta \leq 1/(2\chi)$, it holds that

$$\begin{aligned}
&\Psi_j(c) - \Psi_k(c) \\
&= \left[\mathcal{L}^\theta_c(x^j; p^j) - \mathcal{L}^\theta_c(x^k; p^k)\right] + \frac{a_\theta(\|p^k\|^2 - \|p^j\|^2)}{2\chi c} + \frac{\gamma_\theta(\|\Delta p^j\|^2 - \|\Delta p^k\|^2)}{4B\chi c} \\
&\leq \left[\mathcal{L}^\theta_c(x^j; p^j) - \mathcal{L}^\theta_c(x^k; p^k)\right] + \frac{a_\theta\|p^k\|^2}{2\chi c} + \frac{\gamma_\theta\|\Delta p^j\|^2}{4B\chi c} \\
&\leq \left[\mathcal{L}^\theta_c(x^j; p^j) - \mathcal{L}^\theta_c(x^k; p^k)\right] + \frac{\|p^k\|^2}{2\chi c} + \frac{\|p^{j-1}\|^2 + \|p^j\|^2}{4B\chi^2 c} \\
&\overset{\text{(a)-(b)}}{\leq} \left[\phi(x^j) - \phi(x^k) + \frac{3(\|p^j\|^2 + \|p^{j-1}\|^2)}{\chi^2 c} + \frac{\|p^k\|^2}{2c}\right] + \\
&\qquad \frac{\|p^k\|^2}{2\chi c} + \frac{\|p^{j-1}\|^2 + \|p^j\|^2}{4B\chi^2 c} \leq \Delta_\phi + 4\left(\frac{\|p^j\|^2 + \|p^{j-1}\|^2 + \|p^k\|^2}{\chi^2 c}\right).
\end{aligned}$$ ∎

The next result presents bounds on $S^{(f)}_{j+1,k}$ and $S^{(v)}_{j+1,k}$.

Proposition 3.5. *For $j = 1, \ldots, k-1$,*

$$S^{(f)}_{j+1,k} \leq \frac{\|p^j\| + 2S^{(p)}_{j+1,k}}{\chi c}, \tag{3.10}$$

$$S^{(v)}_{j+1,k} \leq 2\sqrt{\frac{\kappa_0^2 + \kappa_5 c}{k-j}}\left(\Delta_\phi^{1/2} + \frac{\|p^j\| + \|p^{j-1}\| + \|p^k\|}{\chi\sqrt{c}}\right), \tag{3.11}$$

where $(\kappa_0, \kappa_5, \Delta_\phi)$ is as in (2.7).

Proof. Using Lemma 3.1(a), the fact that $\theta \in (0,1)$, and the triangle inequality, it holds that

$$S^{(f)}_{j+1,k} = \frac{\sum_{i=j+1}^k \|p^i - (1-\theta)p^{i-1}\|}{\chi c(k-j)} \leq \frac{\sum_{i=j+1}^k (\|p^{i-1}\| + \|p^i\|)}{\chi c(k-j)} \leq \frac{\|p^j\| + 2S^{(p)}_{j+1,k}}{\chi c},$$

which is (3.10). On the other hand, to show (3.11), we use the definition of $S^{(v)}_{j+1,k}$, the fact that $\sqrt{a+b} \leq \sqrt{a}+\sqrt{b}$ for $a,b \in \mathbb{R}_+$, Lemma 3.3, and Lemma 3.4(c), to conclude that

$$
\begin{aligned}
S^{(v)}_{j+1,k} &= \frac{\sum_{i=j+1}^{k} \|v^i\|}{k-j} \leq \left(\frac{\sum_{i=j+1}^{k} \|v^i\|^2}{k-j} \right)^{1/2} \\
&\overset{\text{L.3.3}}{\leq} \left(\frac{[\kappa_0^2 + \kappa_5 c][\Psi_j(c) - \Psi_k(c)]}{k-j} \right)^{1/2} \\
&\overset{\text{L.3.4(c)}}{\leq} \sqrt{\frac{\kappa_0^2+\kappa_5 c}{k-j}} \left[\Delta_\phi + 4 \left(\frac{\|p^j\|^2 + \|p^{j-1}\|^2 + \|p^k\|^2}{\chi^2 c} \right) \right]^{1/2} \\
&\leq 2 \sqrt{\frac{\kappa_0^2+\kappa_5 c}{k-j}} \left(\Delta_\phi^{1/2} + \frac{\|p^j\| + \|p^{j-1}\| + \|p^k\|}{\chi\sqrt{c}} \right). \qquad (3.12)
\end{aligned}
$$

□

Observe that both residuals $S^{(v)}_{j+1,k}$ and $S^{(f)}_{j+1,k}$ depend on the size of the Lagrange multipliers p^j, p^{j-1}, and p^k. If all the multipliers generated by Algorithm 2.1 could be shown to be bounded independent of c then it would be easy to see that (3.10)–(3.11) with $j=1$ and some $c=\Theta(\eta^{-1})$ would imply the existence of some $k=O(\eta^{-1}\rho^{-2})$ such that $[S^{(v)}_{2,k}/\rho] + [S^{(f)}_{2,k}/\eta] \leq 1$. Consequently, Algorithm 2.1 would find a solution of Problem $\mathcal{S}_{\rho,\eta}$ in $O(\eta^{-1}\rho^{-2})$ iterations.

Unfortunately, we do not know how to bound $\{\|p_i\|\}$ independent of c, so we will instead show the existence of $1 \leq j \leq k$ such that (i) indices j and $k-j$ are $\Theta(\eta^{-1}\rho^{-2})$ and (ii) the three multipliers p^j, p^{j-1}, and p^k are bounded. This fact and Proposition 3.5 suffice to show that the last (hypothetical) conclusion in the previous paragraph actually holds.

3.2. Bounding the Lagrange Multipliers. This subsection generalizes the analysis in [19]. More specifically, Proposition 3.8 shows that if k is sufficiently large relative to an index j, the penalty parameter c, and $\|p^0\|$, then $S^{(p)}_{j+1,k} = \mathcal{O}(1)$.

The proof of the first result can be found in [26, Lemma B.3] using the variable substitution $(q, q^-, \chi) = (q^i, [1-\theta]p^{i-1}, c)$ and step 4 of Algorithm 2.1.

LEMMA 3.6. *For every $i \geq 1$ and $r \in \partial h(z^i) + A^* q^i$, it holds that*

$$
\|q^i\| \leq \max\left\{ (1-\theta)\|p^{i-1}\|, \frac{2D_\dagger(K_h + \|r\|)}{d_\ddagger \sigma_A^+} \right\}.
$$

The next result presents some fundamental properties about p^{i-1}, p^i, and q^i.

LEMMA 3.7. *For every $1 \leq j \leq k$,*

(a) $p^j = \chi q^j + (1-\chi)(1-\theta)p^{j-1}$;

(b) $\|p^j\| \leq \|p^0\| + \kappa_1 c$;

(c) it holds that

$$
\frac{(1-\theta)\|p^k\|}{k-j} + \theta S^{(p)}_{j+1,k} \leq \frac{(1-\theta)\|p^j\|}{k-j} + \frac{2\chi D_\dagger \left[K_h + G_f + S^{(v)}_{j+1,k} \right]}{d_\dagger \sigma_A^+},
$$

where K_h, $d_\dagger$, and $(D_\dagger, G_f)$ are as in (A3), (A6), and (2.4), respectively.

Proof. (a) This is an immediate consequence of the updates for p^j and q^j in Algorithm 2.1.

(b) In view of Step 3 of Algorithm 2.1, the fact that $\theta \in (0,1)$, and the triangle inequality, it holds that

$$\begin{aligned}
\|p^j\| &\le (1-\theta)\|p^{j-1}\| + \chi c\|f^j\| \le (1-\theta)^j\|p^0\| + \chi c\sum_{i=0}^{j-1}(1-\theta)^i\|f^i\| \\
&\le \|p^0\| + \chi c\|A\|\sup_{z\in\mathcal{H}}\|z - z_\dagger\|\sum_{i=0}^{\infty}(1-\theta)^i \\
&= \|p^0\| + \frac{\chi c\|A\|D_\dagger}{\theta} = \|p^0\| + \kappa_1 c.
\end{aligned}$$

(c) Let $i \ge 1$ be fixed, define

$$d_{\chi,\theta} := (1-\theta)(1-\chi),$$

and recall that Lemma 3.1(b) implies $v^i - \nabla f(x^i) \in \partial h(x^i) + A^* q^i$. Using Lemma 3.6 with $r = v^i - \nabla f(x^i)$, the definition of G_f in (2.4), and part (a), we first have that

$$\begin{aligned}
\|p^i\| &\overset{(a)}{=} \|\chi q^i + d_{\chi,\theta}\cdot p^{i-1}\| \le \chi\|q^i\| + d_{\chi,\theta}\|p^{i-1}\| \\
&\overset{\text{L.3.6}}{\le} d_{\chi,\theta}\|p^{i-1}\| + \chi\max\left\{(1-\theta)\|p^{i-1}\|, \frac{2D_\dagger(K_h + \|v^i - \nabla f(x^i)\|)}{d_\dagger\sigma_A^+}\right\} \\
&\le (1-\theta)(1-\chi)\|p^{i-1}\| + \chi\left[(1-\theta)\|p^{i-1}\| + \frac{2D_\dagger(K_h + \|v^i - \nabla f(x^i)\|)}{d_\dagger\sigma_A^+}\right] \\
&\le (1-\theta)\|p^{i-1}\| + \frac{2\chi D_\dagger(K_h + \|\nabla f(x^i)\| + \|v^i\|)}{d_\dagger\sigma_A^+} \\
&\le (1-\theta)\|p^{i-1}\| + \frac{2\chi D_\dagger(K_h + G_f + \|v^i\|)}{d_\dagger\sigma_A^+}.
\end{aligned}$$

Summing the above inequality from $i = j+1$ to k and dividing by $k-j$ yields the desired conclusion. ∎

We are now ready to present the claimed bound on $S^{(p)}_{j+1,k}$.

Proposition 3.8. *Let $\mathcal{R} \ge 0$ and $\underline{c} > 0$ be given and suppose c and p^0 satisfy (2.10). Then, for any positive integers j and k such that $k - j \ge \kappa_6 + \xi^{(0)}_{\mathcal{R}}c + \xi^{(1)}_{\mathcal{R}}c^2$, we have*

$$S^{(p)}_{j+1,k} \le \kappa_2,$$

where (κ_2, κ_6) and $(\xi^{(0)}_{\mathcal{R}}, \xi^{(1)}_{\mathcal{R}})$ are as in (2.7) and (2.9), respectively.

Proof. Using (2.10), (3.11), Lemma 3.7(b), and the relation $\sqrt{a} + \sqrt{b} \le \sqrt{2(a+b)}$ for $a, b \in \mathbb{R}_+$, we first have that

$$\begin{aligned}
S^{(v)}_{j+1,k} &\le 2\sqrt{\frac{\kappa_0^2 + \kappa_5 c}{k-j}}\left(\Delta_\phi^{1/2} + \frac{\|p^j\| + \|p^{j-1}\| + \|p^k\|}{\chi\sqrt{c}}\right) \\
&\le \sqrt{\frac{4(\kappa_0^2 + \kappa_5 c)}{k-j}}\left(\Delta_\phi^{1/2} + \frac{3[\|p^0\| + \kappa_1 c]}{\chi\sqrt{c}}\right)
\end{aligned}$$

$$
\le \sqrt{\frac{4(\kappa_0^2+\kappa_5 c)}{k-j}}\left(\Delta_\phi^{1/2}+\frac{3[\mathcal{R}+\kappa_1]\sqrt{c}}{\chi}\right)
$$

$$
\le \sqrt{\frac{8(\kappa_0^2+\kappa_5 c)}{k-j}\left(\Delta_\phi+\frac{9[\mathcal{R}+\kappa_1]^2 c}{\chi^2}\right)} \le \kappa_4\sqrt{\frac{\xi_{\mathcal{R}}^{(0)}c+\xi_{\mathcal{R}}^{(1)}c^2}{k-j}}.
$$

Using the above bound, Lemma 3.7(b)–(c), our assumed bound on $k-j$, and the definition of κ_2, we conclude that

$$
\begin{aligned}
S_{j+1,k}^{(p)} &\le \frac{2\chi D_\dagger(K_h+G_f)}{\theta d_\dagger \sigma_A^+}+\frac{(1-\theta)\|p^j\|}{\theta(k-j)}+\frac{S_{j+1,k}^{(v)}}{\kappa_4}\\
&\le \frac{2\chi D_\dagger(K_h+G_f)}{\theta d_\dagger \sigma_A^+}+\frac{(1-\theta)(\|p^0\|+\kappa_1 c)}{\theta(k-j)}+\sqrt{\frac{\kappa_6+\xi_{\mathcal{R}}^{(0)}c+\xi_{\mathcal{R}}^{(1)}c^2}{k-j}}\\
&\le \frac{2\chi D_\dagger(K_h+G_f)}{\theta d_\dagger \sigma_A^+}+\frac{(1-\theta)(\mathcal{R}+\kappa_1)c}{\theta(k-j)}+\sqrt{\frac{\kappa_6+\xi_{\mathcal{R}}^{(0)}c+\xi_{\mathcal{R}}^{(1)}c^2}{k-j}}\\
&\le \frac{2\chi D_\dagger(K_h+G_f)}{\theta d_\dagger \sigma_A^+}+\frac{\xi_{\mathcal{R}}^{(0)}c}{\theta(k-j)}+\sqrt{\frac{\kappa_6+\xi_{\mathcal{R}}^{(0)}c+\xi_{\mathcal{R}}^{(1)}c^2}{k-j}}\\
&\le \frac{1}{\theta}\left[1+\frac{2\chi D_\dagger(K_h+G_f)}{\theta d_\dagger \sigma_A^+}\right]+1=\kappa_2.
\end{aligned}
$$

∎

We end this subsection by discussing some implications of the above results. Suppose ζ is an integer satisfying $\zeta \ge \kappa_6+\xi_{\mathcal{R}}^{(0)}c+\xi_{\mathcal{R}}^{(1)}c^2=\Theta(c^2)$. It then follows from Proposition 3.8 that $S_{2,\zeta}^{(p)}=\mathcal{O}(1)$ and $S_{2\zeta,3\zeta}^{(p)}=\mathcal{O}(1)$. Since the minimum of a set of scalars minorizes its average, there exist indices $j_0\in\{2,\ldots,\zeta\}$ and $k_0\in\{2\zeta,\ldots,3\zeta\}$ such that $\|p^{j_0}\|=\mathcal{O}(1)$ and $\|p^{k_0}\|=\mathcal{O}(1)$. Using the fact that $k_0-j_0\ge\zeta$, the above bounds, and (3.10)–(3.11) with $(j,k)=(j_0,k_0)$, it is reasonable to expect that $S_{j_0+1,k_0}^{(f)}=\mathcal{O}(1/c)$ and $S_{j_0+1,k_0}^{(v)}=\mathcal{O}(\sqrt{c/\zeta})$. In the next section, we give the exact steps of this argument and use the resulting bounds to prove Proposition 2.1.

3.3. Proof of Proposition 2.1. Before presenting the proof of Proposition 2.1, we first give two technical results. The first one refines the bounds in Proposition 3.5 using Proposition 3.8, while the second one gives an important implication of (2.12).

LEMMA 3.9. *Let $\mathcal{R}\ge 0$ and $\underline{c}>0$ be given and suppose (c,p^0) satisfies (2.10) for some $\mathcal{R}\ge \prime$ and $\underline{c}>0$. For any integer ζ such that $\zeta\ge\kappa_6+\xi_{\mathcal{R}}^{(0)}c+\xi_{\mathcal{R}}^{(1)}c^2$, there exist $j\in\{3,\ldots,\zeta\}$ and $k\in\{2\zeta+1,\ldots,3\zeta\}$ satisfying*

$$
S_{j+1,k}^{(v)} \le \tilde{\kappa}_{\underline{c}}^{(0)}\sqrt{\frac{\kappa_0^2+\kappa_5 c}{k-j}},\quad S_{j+1,k}^{(f)}\le\frac{6\kappa_2}{\chi c}, \tag{3.13}
$$

where $(\kappa_0,\kappa_2,\kappa_5)$ and $\tilde{\kappa}_0$ is are as in (2.7) and (2.8), respectively.

Proof. Suppose $\zeta\in\mathbb{N}$ satisfies $\zeta\ge\kappa_6+\xi_{\mathcal{R}}^{(0)}c+\xi_{\mathcal{R}}^{(1)}c^2$. Using Proposition 3.8 with $(j,k)=(1,\zeta)$ it holds that there exists $3\le j\le\zeta$ such that

$$
\|p^{j-1}\|+\|p^j\|\le\frac{\sum_{i=3}^{\zeta}(\|p^{i-1}\|+\|p^i\|)}{\zeta-2}\le\frac{2\sum_{i=2}^{\zeta}\|p^i\|}{\zeta-2}
$$

$$= \frac{2(\zeta-1)S^{(p)}_{2,\zeta}}{\zeta-2} \le 4S^{(p)}_{2,\zeta} \le 4\kappa_2. \tag{3.14}$$

On the other hand, using Proposition 3.8 with $(j,k)=(2\zeta,3\zeta)$ it holds that there exists $k\in\{2\zeta+1,\ldots,3\zeta\}$ such that

$$\|p^k\| \le \frac{\sum_{i=2\zeta+1}^{3\zeta}\|p^i\|}{\zeta} = S_{2\zeta+1,3\zeta} \le \kappa_2. \tag{3.15}$$

Combining (3.14), (3.15), and Proposition 3.5, it follows that

$$\begin{aligned}
S^{(v)}_{j+1,k} &\le 2\sqrt{\frac{\kappa_0^2+\kappa_5 c}{k-j}}\left(\Delta_\phi^{1/2}+\frac{\|p^{j_0}\|+\|p^{j_0-1}\|+\|p^{k_0}\|}{\chi\sqrt{c}}\right)\\
&\overset{(3.14)-(3.15)}{\le} 2\sqrt{\frac{\kappa_0^2+\kappa_5 c}{k-j}}\left(\Delta_\phi^{1/2}+\frac{5\kappa_2}{\chi\sqrt{c}}\right)\\
&\le 2\sqrt{\frac{\kappa_0^2+\kappa_5 c}{k-j}}\left(\Delta_\phi^{1/2}+\frac{5\kappa_2}{\chi\sqrt{\underline{c}}}\right) = \tilde{\kappa}^{(0)}_{\underline{c}}\sqrt{\frac{\kappa_0^2+\kappa_5 c}{k-j}},
\end{aligned}$$

which is the first bound in (3.13). To show the other bound in (3.13), we use (3.14) and Proposition 3.8 to conclude that

$$S^{(f)}_{j+1,k} \le \frac{\|p^j\|+2S^{(p)}_{j+1,k}}{\chi c} \le \frac{6\kappa_2}{\chi c}.$$

∎

We now state a technical result which will be used in the proof of Proposition 2.1(c).

LEMMA 3.10. *For any $\mathcal{R}\ge 0$ and $c\ge\underline{c}>0$, the following statements hold:*

(a) the quantity $\mathcal{T}_c(\cdot,\cdot\,|\,\cdot,\cdot)$ defined in (2.11) satisfies

$$\mathcal{T}_c(\rho,\eta\,|\,\underline{c},\mathcal{R}) \le \left[\left(\frac{c}{\underline{c}}\right)^2+\frac{c}{\underline{c}\cdot\min\{\rho^2,\eta^2\}}\right]\mathcal{T}_{\underline{c}}(1,1\,|\,\underline{c},\mathcal{R});$$

(b) if c satisfies (2.12), then $\mathcal{T}_c(\rho,\eta\,|\,\underline{c},\mathcal{R})\le c^3$.

Proof. (a) This statement follows immediately from the definition of $\mathcal{T}_c(\cdot,\cdot\,|\,\cdot,\cdot)$ and the fact that for any $c\ge\bar{c}$ any nonnegative scalars α, β, and γ, we have

$$\alpha+\beta c \le (\alpha+\beta\underline{c})\left(\frac{c}{\underline{c}}\right), \quad \alpha+\beta c+\gamma c^2 \le (\alpha+\beta\underline{c}+\gamma\underline{c}^2)\left(\frac{c}{\underline{c}}\right)^2.$$

(b) Define $\hat{c}:=\hat{c}(\rho,\eta\,|\,\underline{c},\mathcal{R})$, $\varepsilon:=\min\{\rho,\eta\}$, and $T:=\mathcal{T}_{\underline{c}}(1,1\,|\,\underline{c},\mathcal{R})$, and assume that c satisfies (2.12), or equivalently, $c\ge\hat{c}$. To show the conclusion of (b), it suffices to show that

$$\left[\left(\frac{c}{\underline{c}}\right)^2+\frac{c}{\underline{c}\cdot\varepsilon^2}\right]T \le c^3. \tag{3.16}$$

in view of part (a). It is easy to see that the above inequality is satisfied by any c such that

$$c \ge \pi_\varepsilon := \frac{T/\underline{c}^2+\sqrt{T^2/\underline{c}^4+4T/(\varepsilon^2\underline{c})}}{2}.$$

Since the definition of $\hat{c}$ in (2.12) and the relation $\sqrt{a+b} \le \sqrt{a} + \sqrt{b}$ for $a, b \in \mathbb{R}_+$ imply that $\hat{c} \ge \pi_\varepsilon$, the conclusion of (b) follows from the assumption that $c \ge \hat{c}$ and the previous observation. ∎

We now remark on Lemma 3.9. For any integer $\zeta \ge \kappa_6 + \xi_{\mathcal{R}}^{(0)} c + \xi_{\mathcal{R}}^{(1)} c^2$, it follows that there exist $i_1, i_2 \le 3\zeta$ such that $\|v_{i_1}\| = \mathcal{O}(\sqrt{c/\zeta})$ and $\|f_{i_2}\| = \mathcal{O}(1/c)$. Hence, for some $c = \Theta(\eta^{-1})$ and some $\zeta \ge \Omega(\rho^{-2}\eta^{-1})$, we can guarantee that $\|v_{i_1}\| \le \rho$ and $\|f_{i_2}\| \le \eta$. Clearly, if $i_1 = i_2$ then this argument shows that a solution of Problem $\mathcal{S}_{\rho,\eta}$ can be found in $\mathcal{O}(\rho^{-2}\eta^{-1})$ iterations of Algorithm 2.1. In the proof (of Proposition 2.1) below, we give a more involved argument that guarantees that the above i_1 and i_2 can be chosen so that $i_1 = i_2$.

Proof of Proposition 2.1. (a) Let $(\rho, \eta) \in \mathbb{R}^2_{++}$, $p^0 \in A(\mathbb{R}^n)$, and $c > 0$ be given, and define

$$T := \mathcal{T}_c(\rho, \eta \,|\, \underline{c}, \mathcal{R}), \quad r_j := \frac{\mathcal{S}_j^{(v)}}{\rho} + \frac{\mathcal{S}_j^{(f)}}{\eta}\sqrt{\frac{c^3}{j}} \quad \forall j \ge 1,$$

where $\mathcal{S}_j^{(v)}$ and $\mathcal{S}_j^{(f)}$ are as in Step 2b of Algorithm 2.1 and $\mathcal{T}_c(\cdot, \cdot \,|\, \cdot, \cdot)$ is as in (2.11). For the sake of contradiction, suppose that Algorithm 2.1 has not terminated by the end of iteration $k = T$. Since Algorithm 2.1 (see its Step 2b) terminates unsuccessfully at iteration k exactly when $r_k \le 1$, we will obtain the desired contradiction by showing that there exists $k \le T$ such that $r_k \le 1$.

First, consider an arbitrary pair of integers j and k such that $1 \le j \le k \le T$ and assume without loss of generality that k is even. Then, combining (3.18), the relations $S_{k/2,k}^{(v)} = \mathcal{S}_k^{(v)}$ and $S_{k/2,k}^{(f)} = \mathcal{S}_k^{(f)}$, we easily see that

$$\begin{aligned} r_k &= \frac{S_{k/2,k}^{(v)}}{\rho} + \frac{c^{3/2} S_{k/2,k}^{(f)}}{\eta\sqrt{k}} = \frac{k-j+1}{k-k/2+1}\left[\frac{S_{j,k}^{(v)}}{\rho} + \frac{c^{3/2} S_{j,k}^{(f)}}{\eta\sqrt{k}}\right] \\ &\le \frac{k+2}{k/2+1}\left[\frac{S_{j,k}^{(v)}}{\rho} + \frac{c^{3/2} S_{j,k}^{(f)}}{\eta\sqrt{k}}\right] = 2\left[\frac{S_{j,k}^{(v)}}{\rho} + \frac{c^{3/2} S_{j,k}^{(f)}}{\eta\sqrt{k}}\right], \end{aligned} \tag{3.17}$$

We now show that there exists suitable j and k so that the last expression is bounded by 1 and hence that our desired contradiction follows. Note first that the definition of $T = \mathcal{T}_c(\rho, \eta)$ in (2.11) implies that $\zeta := T/3$ satisfies the assumption of Lemma 3.9. Hence, the conclusion of this lemma implies the existence of $j \in \{3, \ldots, T/3\}$ and $k \in \{2T/3 + 1, \ldots, T\}$ such that

$$\begin{aligned} \frac{S_{j,k}^{(v)}}{\rho} + \frac{c^{3/2} S_{j,k}^{(f)}}{\eta\sqrt{k}} &\le \frac{\tilde{\kappa}_{\underline{c}}^{(0)}\sqrt{\kappa_0^2 + \kappa_5 c}}{\rho\sqrt{k-j}} + \frac{6\kappa_2\sqrt{c}}{\chi\eta\sqrt{k}} \le \frac{\tilde{\kappa}_{\underline{c}}^{(0)}\sqrt{\kappa_0^2 + \kappa_5 c}}{\rho\sqrt{T/3}} + \frac{6\kappa_2\sqrt{c}}{\chi\eta\sqrt{T/3}} \\ &= \sqrt{\frac{\tilde{\kappa}_1 + \tilde{\kappa}_2 c}{\rho^2 T}} + \sqrt{\frac{\kappa_3 c}{\eta^2 T}} \le \frac{1}{4} + \frac{1}{4} = \frac{1}{2}, \end{aligned} \tag{3.18}$$

where the last inequality follows from the definition of T. Combining (3.17) and (3.18) we conclude that $r_k \le 1$, which yields our desired contradiction.

(b) This follows immediately from the stopping condition in Step 2a of Algorithm 2.1 and Lemma 3.1(b).

(c) Let (T, r_k) be as in part (a) and assume that c satisfies (2.12). Assume, for contradiction, that Algorithm 2.1 does not terminate successfully. Then, by part (a), the algorithm terminates in an iteration $k \le T$ such that $r_k \le 1$. Using the fact that r_k itself is an average of scalars, there exists $k/2 \le i \le k$ such that

$$\frac{\|v^i\|}{\rho} + \frac{c^{3/2}\|f^i\|}{\eta\sqrt{k}} \le \frac{S^{(v)}_{k/2,k}}{\rho} + \frac{c^{3/2}S^{(f)}_{k/2,k}}{\eta\sqrt{k}} \le 1.$$

Hence, it holds that $\|v^i\| \le \rho$ and $\|f^i\| \le \eta\sqrt{k}c^{-3/2} \le \eta\sqrt{T}c^{-3/2}$ where the last inequality is due to the fact that $k \le T$. Moreover, the assumption that c satisfies (2.12) together with Lemma 3.10(b) then imply that $T \le c^3$ and, hence, that $\|f^i\| \le \eta$. Consequently, this means that the algorithm actually terminates successfully at iteration $i \le k$. We have thus established the desired contradiction and, hence, that part (c) holds. ∎

4. Analysis of Algorithm 2.2. This section presents the main properties of Algorithm 2.2, including the proof of Theorem 2.2.

We first start with two crucial technical results.

PROPOSITION 4.1. *The following statements hold about the ℓ^{th} iteration of Algorithm 2.2:*

(a) $\|\bar{p}^{\ell-1}\|/c_\ell \le 2\kappa_1$, *where κ_1 is as in (2.7);*

(b) its call to Algorithm 2.1 terminates in $\mathcal{T}_{c_\ell}(\rho,\eta\,|\,c_1,2\kappa_1)$ iterations and, if the ℓ^{th} penalty parameter $c_\ell > 0$ satisfies

$$c_\ell \ge \hat{c}(\rho,\eta\,|\,c_1,2\kappa_1), \tag{4.1}$$

then this call terminates successfully, where κ_1, $\mathcal{T}_c(\cdot,\cdot\,|\,\cdot,\cdot)$, and $\hat{c}(\cdot,\cdot\,|\,\cdot,\cdot)$ are as in (2.7), (2.11), and (2.12), respectively.

Proof. (a) We proceed by induction. Since $\bar{p}^0 = 0$, the case of $\ell = 1$ is immediate. Suppose the statement holds for some iteration ℓ and, hence, that $\|\bar{p}^{\ell-1}\| \le 2\kappa_1 c_\ell$. Then, it follows from Lemma 3.7(b) with $(p^0, c) = (\bar{p}^{\ell-1}, c_\ell)$ and the relation $c_{\ell+1} = 2c_\ell$ that

$$\|\bar{p}^\ell\| \le \|\bar{p}^{\ell-1}\| + \kappa_1 c_\ell \le 2\kappa_1 c_\ell + \kappa_1 c_\ell = 3\kappa_1 c_\ell = \frac{3\kappa_1}{2}c_{\ell+1} < 2\kappa_1 c_{\ell+1}.$$

(b) This follows from part (a), the fact that $\{c_\ell\}_{\ell\ge1}$ is an increasing sequence, and Proposition 2.1 with $(c, \underline{c}, \mathcal{R}) = (c_\ell, c_1, 2\kappa_1)$. ∎

We are now ready to give the proof of Theorem 2.2.

Proof of Theorem 2.2. Define the scalars

$$\hat{c} := \hat{c}(\rho,\eta\,|\,c_1,2\kappa_1), \quad \hat{\ell} := \lceil \log_2^+(\hat{c}/c_1)\rceil, \quad \mathcal{T}_{c_\ell} := \mathcal{T}_{c_\ell}(\rho,\eta\,|\,c_1,2\kappa_1),$$

where $\hat{c}(\cdot,\cdot\,|\,\cdot,\cdot)$ is as in (2.12). Proposition 4.1(b) and the update rule for c_ℓ imply that Algorithm 2.2 performs at most $\hat{\ell}$ iterations, and terminates with a pair that solves Problem $\mathcal{S}_{\rho,\eta}$. Moreover, the total number of iterations of Algorithm 2.1 (performed by all of Algorithm 2.2's calls to it) is bounded by $\sum_{\ell=1}^{\hat{\ell}}\mathcal{T}_{c_\ell}$. Now, using Lemma 3.10(a) with $\underline{c} = c_1$, it follows that

$$\frac{\sum_{\ell=1}^{\hat{\ell}}\mathcal{T}_{c_\ell}}{T_1} \le \frac{\sum_{\ell=1}^{\hat{\ell}}c_\ell^2}{c_1^2} + \frac{\sum_{\ell=1}^{\hat{\ell}}c_\ell}{c_1\varepsilon^2} = \sum_{\ell=1}^{\hat{\ell}}2^{2(\ell-1)} + \frac{\sum_{\ell=1}^{\hat{\ell}}2^{(\ell-1)}}{\varepsilon^2} \le 4^{\hat{\ell}} + \frac{2^{\hat{\ell}}}{\varepsilon^2}, \tag{4.2}$$

where (T_1, ε) are as in (2.13). We now derive suitable bounds for $4^{\hat{\ell}}$ and $2^{\hat{\ell}}$. Using the definitions of $\hat{c}$ and $\hat{\ell}$, and the definition of (E_0, E_1) in (2.15), we first have that

$$2^{\hat{\ell}} \leq \max\left\{2, 2^{(1+\log_2 \hat{c}/c_1)}\right\} \leq 2\max\left\{1, \frac{\hat{c}}{c_1}\right\} = 2\max\left\{1, \frac{1}{c_1^3}\left(T_1 + \frac{\sqrt{c_1^3 T_1}}{\varepsilon}\right)\right\}$$

$$(4.3) \qquad \leq 2\left(1 + \frac{T_1}{c_1^3} + \frac{1}{\varepsilon}\sqrt{\frac{T_1}{c_1^3}}\right) = E_0 + \frac{E_1}{\varepsilon}.$$

Combining the above inequality above with the bound $(a+b)^2 \leq 2a^2 + 2b^2$ for $a, b \in \mathbb{R}$, it is also easy to see that

$$(4.4) \qquad 4^{\hat{\ell}} \leq (2^{\hat{\ell}})^2 \leq 2E_0^2 + \frac{2E_1^2}{\varepsilon^2}.$$

The conclusion now follows by applying (4.4) and (4.3) to (4.2). ∎

5. Numerical Experiments.

This section examines the performance of the proposed DP.ADMM (Algorithm 2.2) for finding stationary points of a nonconvex three-block distributed quadratic programming problem. Specifically, given a radius $\gamma > 0$ and a dimension $n \in \mathbb{N}$, it considers the three-block problem

$$\begin{aligned}
\min_{(x_1,x_2,x_3)\in\mathbb{R}^n\times\mathbb{R}^n\times\mathbb{R}^n} \quad & -\sum_{i=1}^{2}\left[\frac{\alpha_i}{2}\|x_i\|^2 + \langle x_i, \beta_i\rangle\right] \\
\text{s.t.} \quad & \|x\|_\infty \leq \gamma, \\
& x_1 - x_3 = 0, \\
& x_2 - x_3 = 0,
\end{aligned}$$

where $\{\alpha_i\}_{i=1}^2 \subseteq [0,1]$, $\{\beta_i\}_{i=1}^2 \subseteq [0,1]^n$, and the entries of these quantities are sampled from the uniform distribution on $[0,1]$. It is clear that the above problem is an instance of (1.1) if we take h_i to be the indicator of the set $\{x \in \mathbb{R}^n : \|x\|_\infty \leq \gamma\}$ for $i = 1, \ldots, 3$. At the end of this section, we give some elucidating remarks.

Before presenting the results, we first describe the algorithms tested. The first set of algorithms, labeled DP1–DP2, are modifications of Algorithm 2.2. Specifically, both DP1 and DP2 replace the original definition of $\mathcal{S}_k^{(f)}$ (resp. $\mathcal{S}_k^{(f)}$) in Step 2b of Algorithm 2.1 with $2\sum_{i=1}^k \|v^i\|/[k+2]$ (resp. $2\sum_{i=1}^k \|Ax^i - d\|/[k+2]$) and choose $(\lambda, c_1) = (1/2, 1)$. Moreover, DP1 chooses $(\theta, \chi) = (0, 1)$ while DP2 chooses $(\theta, \chi) = (1/2, 1/18)$ which satisfies (2.6) at equality. The second set of algorithms, labeled SDD1–SDD3, are instances of the SDD-ADMM of [28] for different values of the penalty parameter ρ. Specifically, all of these instances uses the parameters $(\omega, \theta, \tau) = (4, 2, 1)$, following the same choice as in [28, Section 5.1], and select the following curvature constants: $(M_h, K_h, J_h, L_h) = (4\gamma, 1, 1, 0)$. Moreover, SDD1–SDD3 respectively choose the penalty parameter ρ to be 0.1, 1.0, and 10.0, and termination of the method occurs when the norm of the stationary residual ξ^k and feasibility are both less than a given numerical tolerance.

The results of our experiment are now given in Tables 5.1–5.2, which present both iteration counts and runtimes for either varying choices of γ (Table 5.1) or n (Table 5.2). We now describe a few more details about these experiments and tables. First, the starting point for all methods is the zero vector and the numerical tolerances (e.g., ρ and η in DP1–DP2) for each method were set to be 10^{-9}. Second, the bolded text in the tables highlight the method that performed the best in terms of iteration count. Third, we imposed an iteration limit of 100,000 and marked the runs which did not terminate by this limit with a '-' symbol. Fourth, the experiments were implemented and executed in Matlab R2021b on a Windows 64-bit desktop machine with 12GB of RAM and two Intel(R) Xeon(R) Gold 6240 processors, and the code is readily available online[3].

	Iteration Count					Runtime (ms)				
γ	DP1	DP2	SDD1	SDD2	SDD3	DP1	DP2	SDD1	SDD2	SDD3
10^0	**21**	29	363	135	528	1.8	1.9	38.2	13.4	50.4
10^1	**76**	83	427	223	976	4.0	4.9	41.3	22.4	88.1
10^2	**151**	156	497	309	1394	7.9	7.7	45.2	28.3	121.7
10^3	**228**	232	569	399	1855	10.8	10.8	51.2	34.3	159.3
10^4	**306**	308	647	489	2316	15.5	17.6	58.9	42.9	223.1
10^5	**385**	385	-	581	2778	17.9	18.5	-	48.0	241.5

TABLE 5.1
Results with $n = 10$ and different values of γ

	Iteration Count					Runtime (ms)				
n	DP1	DP2	SDD1	SDD2	SDD3	DP1	DP2	SDD1	SDD2	SDD3
10	**151**	156	497	309	1394	7.8	7.5	65.8	29.0	121.8
40	**55**	60	-	-	3117	3.7	3.5	-	-	319.0
160	**139**	144	-	388	1836	8.5	8.2	-	42.0	202.7
640	**53**	54	-	349	16243	4.0	3.9	-	40.4	1901.5
2560	**58**	59	-	458	8464	7.1	6.7	-	77.4	1553.7
10240	**108**	110	-	1058	4334	44.4	40.3	-	623.5	2790.6

TABLE 5.2
Results with $\gamma = 100$ and different values of n

From the results in Tables 5.1–5.2, we see that DP1 performed the best in terms of iteration count and DP2 had iteration counts that were close to DP1. On the other hand, SDD2 outperformed its other SDD-ADMM variant on all problems except one. Finally, notice that the DP.ADMM variants scaled better against the dimension n compared to the SDD-ADMM variants.

To close this section, we give some elucidating remarks. First, we excluded the algorithm in [15] due to its poor iteration complexity bound and the fact that it is an algorithm applied to a reformulation of (1.1) rather than to (1.1) directly. Second, we had to choose different values of the penalty parameter ρ for the SDD-ADMM variants because the analysis in [28] did not present a practical way of adaptively updating ρ (note that the "adaptive" method in [28, Algorithm 3.2] is not practical because it requires an estimate of $\sup_{x\in\mathcal{H}} \phi(x) - \inf_{x\in\mathcal{H}} \phi$ for (1.1)).

6. Concluding Remarks.

The analysis of this paper also applies to instances of (1.1) where f is not necessarily differentiable on $\mathcal{H}$ as in our condition (A5), but instead satisfies a more relaxed version of (A5), namely: for every $x \in \mathcal{H}$, the function $f(x_{<t}, \cdot, x_{>t})$ has a Fréchet subgradient at x_t, denoted by $\nabla_{x_t} f(x_{\le t}, x_{>t})$, and (2.3) is satisfied for every $t = 1, \ldots, B-1$. Hence, our analysis immediately applies to

[3]See https://github.com/wwkong/nc_opt/tree/master/tests/papers/dp_admm.

the case where $f(z)$ is of the form $\sum_{t=1}^{B} f_t(z_t)$ in which, for every $t = 1, \ldots, B$, the function $f_t(\cdot) + m_t\|\cdot\|^2/2 + \delta_{\mathcal{H}_t}(\cdot)$ is convex and has a subgradient everywhere in $\mathcal{H}_t$.

We now discuss some possible extensions of our analysis in this paper. First, our analysis was done under the assumption that $\mathcal{H}$ is bounded (see (A3)), but it is straightforward to see that it is still valid under the weaker assumption that $\sup_{k\geq 1}\|x^k - z_\dagger\| \leq D_\dagger$ for some $D_\dagger > 0$ where $z_\dagger$ is as in (A6). It would be interesting to extend the analysis in this paper to the case where $\mathcal{H}$ is unbounded, possibly by assuming conditions on the sublevel sets of ϕ which guarantee that the aforementioned bound holds. Second, the convergence of Algorithm 2.2 is established under the assumption that exact solutions to the subproblems in Step 1 of Algorithm 2.1 are easy to obtain. We believe that convergence can also be established when only inexact solutions, e.g.,

$$x_t^k \approx \operatorname*{argmin}_{u_t\in\mathbb{R}^{n_t}} \left\{ \lambda \mathcal{L}_c^\theta(x_{<t}^k, u_t, x_{>t}^{k-1}; p^{k-1}) + \frac{1}{2}\|u_t - x_t^{k-1}\|^2 \right\} \tag{6.1}$$

are available. For example, one could consider applying an accelerated composite gradient (ACG) method to the problem associated with (6.1) so that x_t^k satisfies

$$\exists r_t^k \quad \text{s.t.} \quad \begin{cases} r_k^t \in \partial\left(\lambda\mathcal{L}_c^\theta(x_{<t}^k, \cdot, x_{>t}^{k-1}; p^{k-1}) + \frac{1}{2}\|\cdot - x_t^{k-1}\|^2\right)(x_t^k), \\ \|r_t^k\|^2 \leq \sigma^2\|x_t^{k-1} - x^k\|^2, \end{cases}$$

for some $\sigma \in (0, 1)$.

Appendix A. Proof of Lemma 3.2 and Lemma 3.4(a)–(b).

Before giving the proofs, we present some auxiliary results. To avoid repetition, we assume the reader is already familiar with (3.1)–(3.3).

The proof of the first result can be found in [19, Lemma B.2].

Lemma A.1. *For any $(\zeta, \theta) \in [0, 1]^2$ satisfying $\zeta \leq \theta^2$ and $(a, b) \in \mathbb{R}^n \times \mathbb{R}^n$, we have that*

$$\|a - (1-\theta)b\|^2 - \zeta\|a\|^2 \geq \left[\frac{(1-\zeta) - (1-\theta)^2}{2}\right]\left(\|a\|^2 - \|b\|^2\right). \tag{A.1}$$

The next result establishes some general bounds given by the updates in (1.5).

Lemma A.2. *For every $i \geq 1$, index $t = 1, \ldots, B$, and $u_t \in \mathcal{H}_t$, it holds that*

$$\begin{aligned}
&\lambda\left[\mathcal{L}_c^\theta(x_{<t}^i, u_t, x_{>t}^{i-1}; p^{i-1}) - \mathcal{L}_c^\theta(x_{<t}^i, x_t^i, x_{>t}^{i-1}; p^{i-1})\right] + \frac{1}{2}\|u_t - x_t^{i-1}\|^2 \\
&\geq \frac{1}{2}\|\Delta x_t^i\|^2 + \left(\frac{1-\lambda m_t}{2}\right)\|u_t - x_t^i\|^2 + \frac{\lambda c}{2}\|A_t(u_t - x_t^i)\|^2.
\end{aligned}$$

Proof. Let $i \geq 1$, $t = 1, \ldots, B$, and $u_t \in \mathcal{H}_t$ be fixed, and define $\mu := 1 - \lambda m_t$ and $\|\cdot\|_\alpha^2 := \langle \cdot, (\mu I + \lambda c A_t^* A_t)(\cdot)\rangle$. Since the prox stepsize λ is chosen in $(0, 1/(2m)]$ and $m \geq m_t$ in view of (2.7), it follows that $\mu \geq 1/2$. Using the optimality of x_t^i, assumption (A4), and the fact that $\lambda\mathcal{L}_c^\theta(x_{<t}^i, \cdot, x_{>t}^{i-1}; p^{i-1}) + \|\cdot - x_t^{i-1}\|^2/2$ is 1-strongly convex with respect to $\|\cdot\|_\alpha^2$, it follows that

$$\begin{aligned}
&\lambda\mathcal{L}_c^\theta(x_{<t}^i, x_t^i, x_{>t}^{i-1}; p^{i-1}) + \frac{1}{2}\|\Delta x_t^i\|^2 \\
&\leq \lambda\mathcal{L}_c^\theta(x_{<t}^i, u_t, x_{>t}^{i-1}; p^{i-1}) + \frac{1}{2}\|u_t - x_t^{i-1}\|^2 - \frac{1}{2}\|u_t - x_t^i\|_\alpha^2 \\
&= \lambda\mathcal{L}_c^\theta(x_{<t}^i, u_t, x_{>t}^{i-1}; p^{i-1}) + \frac{1}{2}\|u_t - x_t^{i-1}\|^2 - \frac{\mu}{2}\|u_t - x_t^i\|^2 - \frac{\lambda c}{2}\|A_t(u_t - x_t^i)\|^2. \quad \square
\end{aligned}$$

We are now ready to give the proof of Lemma 3.2.

Proof of Lemma 3.2. (a) Using the definition of $\mathcal{L}_c^\theta(\cdot;\cdot)$ in (1.4) and the relation in Lemma 3.1(a), we conclude that

$$\begin{aligned}
\mathcal{L}_c^\theta(x^i;p^i) - \mathcal{L}_c^\theta(x^i;p^{i-1}) &= (1-\theta)\left\langle \Delta p^i, f^i \right\rangle = \left(\frac{1-\theta}{\chi c}\right)\|\Delta p^i\|^2 + \frac{a_\theta}{\chi c}\left\langle \Delta p^i, p^{i-1}\right\rangle \\
&= \left(\frac{1-\theta}{\chi c}\right)\|\Delta p^i\|^2 + \frac{a_\theta}{\chi c}\left(\left\langle p^i, p^{i-1}\right\rangle - \|p^{i-1}\|^2\right) \\
&= \left(\frac{1-\theta}{\chi c}\right)\|\Delta p^i\|^2 + \frac{a_\theta}{\chi c}\left(\frac{1}{2}\|p^i\|^2 - \frac{1}{2}\|\Delta p^i\|^2 - \frac{1}{2}\|p^{i-1}\|^2\right) \\
&= \frac{b_\theta}{2\chi c}\|\Delta p^i\|^2 + \frac{a_\theta}{2\chi c}\left(\|p^i\|^2 - \|p^{i-1}\|^2\right). \qquad \text{(A.2)}
\end{aligned}$$

(b) Using the definition of m in (2.7) and summing the inequality of Lemma A.2 with $u_t = x_t^{i-1}$ from $t=1$ to B, we have that

$$\begin{aligned}
\left(1 - \frac{\lambda m}{2}\right)\|\Delta x^i\|^2 + \frac{\lambda c}{2}\sum_{t=1}^B \|A_t \Delta x_t^i\|^2 &\le \sum_{i=1}^t \left(1 - \frac{\lambda m_t}{2}\right)\|\Delta x_t^i\|^2 + \frac{\lambda c}{2}\sum_{t=1}^B \|A_t \Delta x_t^i\|^2 \\
&\le \lambda\left[\mathcal{L}_c^\theta(x^{i-1};p^{i-1}) - \mathcal{L}_c^\theta(x^i;p^{i-1})\right].
\end{aligned}$$

The conclusion now follows from dividing the above inequality by λ and using the fact that $\lambda \le 1/m$.

(c) Note that the definition of b_θ in (3.1) and (2.6) imply

$$\zeta := 2B\chi b_\theta \le \theta^2.$$

Hence, using the definition of γ_θ in (3.1), and Lemma A.1 with $(a,b) = (\Delta p^i, \Delta p^{i-1})$ it follows that

$$\|\Delta p^i - (1-\theta)\Delta p^{i-1}\|^2 \ge 2B\chi b_\theta \|\Delta p^i\|^2 + \chi\gamma_\theta\left(\|\Delta p^i\|^2 - \|\Delta p^{i-1}\|^2\right). \qquad \text{(A.3)}$$

Using (A.3) at i and $i-1$, Lemma 3.1(a), and the relation $\|a\|_1^2 \le n\|a\|_2^2$ for $a \in \mathbb{R}^n$, we have that

$$\begin{aligned}
\frac{c}{4}\sum_{t=1}^B \|A_t \Delta x_t^i\|^2 \ge \frac{c}{4B}\|A\Delta x^i\|^2 &= \frac{\|\Delta p^i - (1-\theta)\Delta p^{i-1}\|^2}{4B\chi^2 c} \\
&\ge \frac{1}{4B\chi c}\left[2Bb_\theta\|\Delta p^i\|^2 + \gamma_\theta\left(\|\Delta p^i\|^2 - \|\Delta p^{i-1}\|^2\right)\right] \\
&= \frac{b_\theta}{2\chi c}\|\Delta p^i\|^2 + \frac{\gamma_\theta}{4B\chi c}\left(\|\Delta p^i\|^2 - \|\Delta p^{i-1}\|^2\right).
\end{aligned}$$

∎

Next, we give the proof of Lemma 3.4(a)–(b).

Proof of Lemma 3.4(a)–(b). (a) Using Lemma 3.2(a), the definition of $\mathcal{L}_c^\theta(\cdot;\cdot)$ in (1.4), the fact that $\theta \in (0,1)$, and the relations $2\langle a,b\rangle \le \|a\|^2 + \|b\|^2$ and $\|a+b\|^2 \le 2\|a\|^2 + 2\|b\|^2$ for $a,b \in \mathbb{R}^n$, it follows that

$$\begin{aligned}
\mathcal{L}_c^\theta(x^j;p^j) &= \phi(x^j) + (1-\theta)\left\langle p^i, f^i\right\rangle + \frac{c}{2}\|f^i\|^2 \\
&\overset{\text{L.3.2(a)}}{=} \frac{(1-\theta)}{\chi c}\left\langle p^i, p^i - (1-\theta)p^{i-1}\right\rangle + \frac{1}{2c\chi^2}\|p^i - (1-\theta)p^{i-1}\|^2
\end{aligned}$$

$$\leq \frac{(1-\theta)}{2\chi c}\|p^i\|^2 + \frac{(1-\theta)}{2\chi c}\|p^i-(1-\theta)p^{i-1}\|^2 + \frac{1}{2\chi^2 c}\|p^i-(1-\theta)p^{i-1}\|^2$$

$$\leq \frac{1}{2\chi c}\|p^i\|^2 + \frac{1}{\chi^2 c}\|p^i-(1-\theta)p^{i-1}\|^2$$

$$\leq \frac{1}{2\chi c}\|p^i\|^2 + \frac{2}{\chi^2 c}\|p^i\|^2 + \frac{2}{\chi^2 c}\|p^{i-1}\|^2 \leq \frac{3(\|p^i\|^2+\|p^{i-1}\|^2)}{\chi^2 c}.$$

(b) It holds that

$$\begin{aligned}\mathcal{L}_c^\theta(x^k;p^k) &= \phi(x^k) + (1-\theta)\left\langle p^k, f^k\right\rangle + \frac{c}{2}\|f^k\|^2 \\ &= \phi(x^k) + \frac{1}{2}\left\|\frac{(1-\theta)p^k}{\sqrt{c}} + \sqrt{c}f^k\right\|^2 - \frac{(1-\theta)^2\|p^k\|^2}{2c} \\ &\geq \phi(x^k) - \frac{(1-\theta)^2\|p^k\|^2}{2c} \geq \phi(x^k) - \frac{\|p^k\|^2}{2c}.\end{aligned}$$

∎

REFERENCES

[1] D. P. Bertsekas, *Nonlinear programming*, Taylor & Francis, 3ed ed., 2016.
[2] S. Boyd, N. Parikh, and E. Chu, *Distributed optimization and statistical learning via the alternating direction method of multipliers*, Now Publishers Inc, 2011.
[3] M. T. Chao, Y. Zhang, and J. B. Jian, *An inertial proximal alternating direction method of multipliers for nonconvex optimization*, Int. J. Comput. Math., (2020), pp. 1–19.
[4] C. Chen, B. He, Y. Ye, and X. Yuan, *The direct extension of admm for multi-block convex minimization problems is not necessarily convergent*, Math. Program., 155 (2016), pp. 57–79.
[5] J. Eckstein and D. P. Bertsekas, *On the douglas–rachford splitting method and the proximal point algorithm for maximal monotone operators*, Math. Program., 55 (1992), pp. 293–318.
[6] J. Eckstein and M. C. Ferris, *Operator-splitting methods for monotone affine variational inequalities, with a parallel application to optimal control*, INFORMS J. Comput., 10 (1998), pp. 218–235.
[7] J. Eckstein and M. Fukushima, *Some reformulations and applications of the alternating direction method of multipliers*, in Large scale optimization, Springer, 1994, pp. 115–134.
[8] J. Eckstein and B. F. Svaiter, *A family of projective splitting methods for the sum of two maximal monotone operators*, Math. Program., 111 (2008), pp. 173–199.
[9] J. Eckstein and B. F. Svaiter, *General projective splitting methods for sums of maximal monotone operators*, SIAM J. Control Optim., 48 (2009), pp. 787–811.
[10] D. Gabay, *Applications of the method of multipliers to variational inequalities*, in Studies in mathematics and its applications, vol. 15, Elsevier, 1983, pp. 299–331.
[11] D. Gabay and B. Mercier, *A dual algorithm for the solution of nonlinear variational problems via finite element approximation*, Comput. Math. Appl., 2 (1976), pp. 17–40.
[12] R. Glowinski and A. Marroco, *Sur l'approximation, par éléments finis d'ordre un, et la résolution, par pénalisation-dualité d'une classe de problèmes de dirichlet non linéaires*, ESAIM Math. Model. Numer. Anal., 9 (1975), pp. 41–76.
[13] M. L. N. Goncalves, J. G. Melo, and R. D. C. Monteiro, *Convergence rate bounds for a proximal ADMM with over-relaxation stepsize parameter for solving nonconvex linearly constrained problems*, Pac. J. Optim., 15 (2019), pp. 379–398.
[14] Z. Jia, J. Huang, and Z. Wu, *An incremental aggregated proximal ADMM for linearly constrained nonconvex optimization with application to sparse logistic regression problems*, J. Comput. Appl. Math., 390 (2021), p. 113384.
[15] B. Jiang, T. Lin, S. Ma, and S. Zhang, *Structured nonconvex and nonsmooth optimization: algorithms and iteration complexity analysis*, Comput. Optim. Appl., 72 (2019), pp. 115–157.
[16] W. Kong, *Accelerated inexact first-order methods for solving nonconvex composite optimization problems*, arXiv preprint arXiv:2104.09685, (2021).
[17] W. Kong, J. G. Melo, and R. D. C. Monteiro, *Complexity of a quadratic penalty accelerated inexact proximal point method for solving linearly constrained nonconvex composite programs*, SIAM J. Optim., 29 (2019), pp. 2566–2593.

[18] W. Kong, J. G. Melo, and R. D. C. Monteiro, *An efficient adaptive accelerated inexact proximal point method for solving linearly constrained nonconvex composite problems*, Comput. Optim. Appl., 76 (2020), pp. 305–346.

[19] W. Kong and R. D. C. Monteiro, *An accelerated inexact dampened augmented Lagrangian method for linearly-constrained nonconvex composite optimization problems*, arXiv preprint arXiv:2110.11151, (2021).

[20] J. G. Melo and R. D. C. Monteiro, *Iteration-complexity of a Jacobi-type non-euclidean ADMM for multi-block linearly constrained nonconvex programs*, arXiv preprint arXiv:1705.07229, (2017).

[21] J. G. Melo and R. D. C. Monteiro, *Iteration-complexity of a linearized proximal multiblock ADMM class for linearly constrained nonconvex optimization problems*, Optimization Online preprint, (2017).

[22] J. G. Melo, R. D. C. Monteiro, and W. Kong, *Iteration-complexity of an inner accelerated inexact proximal augmented Lagrangian method based on the classical Lagrangian function and a full lagrange multiplier update*, arXiv preprint arXiv:2008.00562, (2020).

[23] R. D. C. Monteiro and B. F. Svaiter, *Iteration-complexity of block-decomposition algorithms and the alternating direction method of multipliers*, SIAM J. Optim., 23 (2013), pp. 475–507.

[24] R. T. Rockafellar, *Augmented Lagrangians and applications of the proximal point algorithm in convex programming*, Math. Oper. Res., 1 (1976), pp. 97–116.

[25] A. Ruszczyński, *An augmented Lagrangian decomposition method for block diagonal linear programming problems*, Oper. Res. Lett., 8 (1989), pp. 287–294.

[26] A. Sujanani and R. D. C. Monteiro, *An adaptive superfast inexact proximal augmented Lagrangian method for smooth nonconvex composite optimization problems*, arXiv preprint arXiv:2207.11905, (2022).

[27] A. X. Sun, D. T. Phan, and S. Ghosh, *Fully decentralized ac optimal power flow algorithms*, in 2013 IEEE Power & Energy Society General Meeting, IEEE, 2013, pp. 1–5.

[28] K. Sun and A. Sun, *Dual descent ALM and ADMM*, arXiv preprint arXiv:2109.13214, (2021).

[29] K. Sun and X. A. Sun, *A two-level distributed algorithm for general constrained non-convex optimization with global convergence*, arXiv preprint arXiv:1902.07654, (2019).

[30] A. Themelis and P. Patrinos, *Douglas–rachford splitting and ADMM for nonconvex optimization: Tight convergence results*, SIAM J. Optim., 30 (2020), pp. 149–181.

[31] Y. Wang, W. Yin, and J. Zeng, *Global convergence of ADMM in nonconvex nonsmooth optimization*, J. Sci. Comput., 78 (2019), pp. 29–63.

[32] J. Zhang and Z.-Q. Luo, *A proximal alternating direction method of multiplier for linearly constrained nonconvex minimization*, SIAM J. Optim., 30 (2020), pp. 2272–2302.